



Red Hat Fuse 7.5

Fuse Online のアプリケーションおよびサービス への接続

提供されるコネクタの使用手順

Red Hat Fuse 7.5 Fuse Online のアプリケーションおよびサービスへの接続

提供されるコネクタの使用手順

法律上の通知

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

ステップごとの手順にしたがって、コネクションを作成し、インテグレーションに追加します。

目次

前書き	5
第1章 FUSE ONLINE によってサポートされるコネクタ	7
第2章 AMAZON WEB SERVICES への接続	10
2.1. FUSE ONLINE コネクションを作成するための AWS クレデンシャルの取得	10
2.2. AMAZON SIMPLE NOTIFICATION SERVICE (SNS) への接続	11
2.3. AMAZON SIMPLE QUEUE SERVICE (SQS) への接続	13
2.4. AMAZON SIMPLE STORAGE SERVICE (S3) への接続	17
2.5. AMAZON DYNAMODB への接続	22
第3章 AMQ への接続	27
3.1. AMQ コネクションの作成	27
3.2. AMQ コネクションを追加してメッセージ受信時にインテグレーションの実行をトリガー	28
3.3. AMQ メッセージをパブリッシュしてインテグレーションを終了	30
3.4. インテグレーションの途中で AMQ メッセージを公開	31
第4章 AMQP への接続	34
4.1. AMQP コネクションの作成	34
4.2. AMQP コネクションを追加してメッセージ受信時にインテグレーションの実行をトリガー	35
4.3. フローの途中で AMQP へメッセージをパブリッシュまたは AMQP へメッセージをパブリッシュしてインテグレーションを終了	37
第5章 BOX への接続	40
5.1. FUSE ONLINE を BOX クライアントとして登録	40
5.2. BOX コネクションの作成	41
5.3. インテグレーションの途中での BOX ファイルのダウンロードまたはアップロード	42
5.4. BOX にファイルをアップロードしてインテグレーションを終了	43
第6章 DROPBOX への接続	45
6.1. FUSE ONLINE を DROPBOX クライアントとして登録	45
6.2. DROPBOX コネクションの作成	46
6.3. DROPBOX からファイルを取得してインテグレーションの実行をトリガー	47
6.4. DROPBOX にファイルを追加してインテグレーションを終了	48
6.5. インテグレーションの途中で DROPBOX にアクセス	49
第7章 メールサーバーへの接続 (IMAP、POP3、SMTP)	52
7.1. IMAP または POP3 メールサーバーへのコネクションの作成	52
7.2. SMTP メールサーバーへのコネクションの作成	53
7.3. メールを取得してインテグレーションの実行をトリガー	54
7.4. インテグレーションからのメールの送信	56
第8章 FHIR への接続	58
8.1. FHIR サーバーへのコネクションの作成	58
8.2. FHIR サーバーからのリソースの取得	59
8.3. リソースの FHIR サーバーのクエリー	60
8.4. FHIR サーバーでのリソースの作成	61
8.5. FHIR サーバー上のリソースにあるすべてのフィールドを更新	62
8.6. FHIR サーバー上のリソースにある指定のフィールドを更新	64
8.7. FHIR サーバーで異なるタイプのリソースの作成	65
8.8. FHIR サーバーからのリソースの削除	66
第9章 FTP または SFTP サーバーへの接続	68
9.1. FTP または SFTP コネクションの作成	68

9.2. FTP または SFTP サーバーからのファイルの取得	69
9.3. FTP または SFTP サーバーへのファイルのアップロード	71
第10章 GOOGLE アプリケーションへの接続	74
10.1. FUSE ONLINE を GOOGLE クライアントアプリケーションとして登録	74
10.2. GMAIL への接続	77
10.3. GOOGLE カレンダーへの接続	81
10.4. GOOGLE スプレッドシートへの接続	87
第11章 HTTP および HTTPS エンドポイントへの接続	104
11.1. HTTP または HTTPS エンドポイントへの接続の作成	104
11.2. HTTP または HTTPS 接続のインテグレーションへの追加	104
第12章 IRC への接続	107
12.1. IRC 接続の作成	107
12.2. メッセージの受信時にインテグレーションの実行をトリガーするために IRC 接続を追加	108
12.3. IRC 接続を追加してメッセージを送信	109
第13章 JIRA への接続	111
13.1. OAUTH を使用する JIRA サーバーで FUSE ONLINE を登録	111
13.2. JIRA 接続の作成	115
13.3. JIRA の課題またはコメントを取得してインテグレーションの実行をトリガー	116
13.4. 課題を JIRA サーバーに追加	117
13.5. JIRA の課題へのコメントの追加	118
13.6. JIRA の課題へのファイルの添付	119
13.7. JIRA 課題のウォッチャーの追加および削除	120
13.8. JIRA サーバーでの課題の更新	120
13.9. JIRA の課題の新しいステータスへのトランジション	121
第14章 KAFKA への接続	123
14.1. KAFKA ブローカーへの接続の作成	123
14.2. KAFKA ブローカーからデータを取得してインテグレーションの実行をトリガー	124
14.3. KAFKA ブローカーへのデータの公開	125
第15章 KNATIVE リソースへの接続	127
15.1. KNATIVE リソースへの接続を有効化	127
15.2. KNATIVE チャンネルにサブスクライブしてメッセージを取得	128
15.3. KNATIVE チャンネルへのメッセージの送信	129
15.4. シンプルなインテグレーションを KNATIVE サービスとして公開	131
15.5. シンプルなインテグレーションからの KNATIVE サービスの呼び出し	132
第16章 APACHE KUDU への接続	134
16.1. APACHE KUDU データストアへの接続の作成	134
16.2. スキャンが KUDU テーブルからレコードを返すときにインテグレーションをトリガー	134
16.3. KUDU テーブルへのレコードの挿入	135
第17章 MONGODB への接続	137
17.1. MONGODB データベースへの接続の作成	137
17.2. MONGODB データベースの更新時にインテグレーションをトリガー	138
17.3. MONGODB データベースのドキュメントの操作	140
第18章 MQTT への接続	142
18.1. MQTT ブローカーへの接続の作成	142
18.2. MQTT ブローカーからメッセージを取得してインテグレーションの実行をトリガー	143
18.3. MQTT ブローカーへのメッセージの公開	144

第19章 ODATA への接続	146
19.1. ODATA サービスへの接続の作成	146
19.2. ポーリングが ODATA からデータを返すときにインテグレーションをトリガー	147
19.3. ODATA サービスによって管理されるデータの更新、作成、および削除	149
第20章 REST API への接続	151
20.1. FUSE ONLINE を REST API クライアントとして登録	151
20.2. REST API クライアント接続の作成	152
20.3. API クライアントのインテグレーションへの追加	153
第21章 SALESFORCE への接続	154
21.1. FUSE ONLINE を SALESFORCE クライアントアプリケーションとして登録	154
21.2. SALESFORCE 接続の作成	155
21.3. SALESFORCE 接続のインテグレーションへの追加	156
第22章 SAP CONCUR への接続	158
22.1. SAP CONCUR へ接続するためのユーザーロール	158
22.2. SAP CONCUR 実装サイト用の OAUTH クレデンシャルの取得方法	158
22.3. SAP CONCUR 本番サイト用の OAUTH クレデンシャルの取得方法	159
22.4. SAP CONCUR コネクターの設定	159
22.5. SAP CONCUR 接続の作成	160
22.6. SAP CONCUR 接続のインテグレーションへの追加	161
22.7. インテグレーションデータフィールドを SAP CONCUR フィールドにマップする例	162
第23章 SERVICENOW への接続	163
23.1. SERVICENOW 接続の作成	163
23.2. SERVICENOW からレコードを取得してインテグレーションの実行をトリガー	164
23.3. SERVICENOW でのインポートセットの作成	165
23.4. インテグレーションの間またはインテグレーションを終了するためにレコードを SERVICENOW にコピー	166
23.5. SALESFORCE から SERVICENOW へのインテグレーションの例	167
第24章 SLACK への接続	169
24.1. SLACK 接続の作成	169
24.2. メッセージの受信時に SLACK 接続を追加してインテグレーションの実行をトリガー	170
24.3. SLACK 接続を追加してメッセージを SLACK チャネルまたはユーザーに送信	171
第25章 SQL データベースへの接続	173
25.1. データベース接続の作成	173
25.2. データベース接続でスキーマを指定する方法	174
25.3. データベースレコードを取得してインテグレーションの実行をトリガー	176
25.4. データを更新する SQL ステートメントのパラメータープレースホルダーおよび値	177
25.5. インテグレーションの途中またはインテグレーションを完了するためにデータベースにアクセス	178
25.6. プロプライエタリーデータベースへの接続	179
第26章 TELEGRAM への接続	181
26.1. TELEGRAM 接続の作成	181
26.2. メッセージの受信時に TELEGRAM 接続を追加してインテグレーションの実行をトリガー	182
26.3. メッセージを TELEGRAM チャットに送信するため TELEGRAM 接続を追加	182
第27章 TWITTER への接続	184
27.1. FUSE ONLINE を TWITTER クライアントアプリケーションとして登録	184
27.2. TWITTER 接続の作成	185
27.3. TWITTER 接続を追加してインテグレーションの実行をトリガー	186
27.4. ダイレクトメッセージを送信する TWITTER 接続の追加	188

前書き

アプリケーションを統合するには、統合する各アプリケーションまたはサービスへのコネクションを作成します。コネクションの作成後にインテグレーションを作成し、統合するインテグレーションまたはサービスごとにコネクションを追加します。

Fuse Online は、コネクションを作成するためのテンプレートとなる [さまざまなコネクタをサポートします](#)。以下のトピックでは、コネクションを作成し、インテグレーションに追加するための詳細を説明しています。

- [2章Amazon Web Services への接続](#)
- [3章AMQ への接続](#)
- [4章AMQP への接続](#)
- [5章Box への接続](#)
- [6章Dropbox への接続](#)
- [7章メールサーバーへの接続 \(IMAP、POP3、SMTP\)](#)
- [8章FHIR への接続](#)
- [9章FTP または SFTP サーバーへの接続](#)
- [10章Google アプリケーションへの接続](#)
- [11章HTTP およびHTTPS エンドポイントへの接続](#)
- [12章IRC への接続](#)
- [13章Jira への接続](#)
- [14章Kafka への接続](#)
- [15章Knative リソースへの接続](#)
- [16章Apache Kudu への接続](#)
- [17章MongoDB への接続](#)
- [18章MQTT への接続](#)
- [19章OData への接続](#)
- [20章REST API への接続](#)
- [21章Salesforce への接続](#)
- [22章SAP Concur への接続](#)
- [23章ServiceNow への接続](#)
- [24章Slack への接続](#)
- [25章SQL データベースへの接続](#)

- [26章 Telegram への接続](#)
- [27章 Twitter への接続](#)

第1章 FUSE ONLINE によってサポートされるコネクタ

Fuse Online では、以下のコネクタがサポートされます。

名前	説明
Amazon DynamoDB	Amazon DynamoDB テーブルでデータを取得、追加、更新、または削除します。
Amazon Simple Notification Service (SNS)	メッセージを Amazon SNS トピックに送信します。
Amazon Simple Queue Service (SQS)	Amazon SQS キューからメッセージを取得したり、Amazon SQS キューにメッセージを送信したりします。
Amazon Simple Storage Service (S3)	Amazon S3 バケットからデータを取得したり、データを Amazon S3 バケットにコピーしたりします。
AMQ	Red Hat AMQ (Apache ActiveMQ) ブローカーからメッセージを取得する Red Hat AMQ (Apache ActiveMQ) ブローカーにメッセージを公開します。
AMQP	AMQP (Advanced Message Queue Protocol) ブローカーからメッセージを取得するか、AMQP ブローカーにメッセージを公開します。
Box	Box からファイルをダウンロードしたり、Box にファイルをアップロードしたりします。
Dropbox	Dropbox からファイルをダウンロードしたり、Dropbox にファイルをアップロードしたりします。
メールサーバー	IMAP または POP3 メールサーバーからメールを取得したり、SMTP メールサーバーに接続してメールを送信したりします。
FHIR	FHIR サーバーからリソースを取得したり、FHIR サーバーのリソースを更新したりします。
FTP/SFTP	FTP または SFTP サーバーからファイルをダウンロードしたり、FTP または SFTP サーバーへファイルをアップロードしたりします。
Gmail	特定の Gmail アカウントへ送信されたメッセージを取得し、特定の Gmail アカウントからメッセージを送信します。
Google カレンダー	イベントを指定する Google カレンダーから取得したり、指定する Google カレンダーのイベントを追加または更新したりします。
Google スプレッドシート	指定する Google スプレッドシートからデータを取得、スプレッドシートのデータを追加/更新、チャートを作成、または指定するスプレッドシートでピボットテーブルを作成します。

名前	説明
HTTP/HTTPS	HTTP または HTTPS エンドポイントに接続し、 GET 、 PUT 、 POST 、 DELETE 、 HEAD 、 OPTIONS 、 TRACE 、または PATCH メソッドを実行します。
IRC	IRC ニックネームまたはチャンネルに送信されたメッセージを受信したり、特定のチャンネルの IRC ニックネームにメッセージを送信したりします。
Jira	Jira サーバーで課題を取得、作成、または更新します。
Knative resources	Knative チャンネルからメッセージを取得、Knative チャンネルへメッセージを送信、シンプルなインテグレーションを Knative サービスとして公開、または Knative サービスを呼び出します。
Kudu	Apache Kudu データストアのテーブルからレコードを取得したり、Kudu データストアのテーブルにレコードを追加したりします。
MongoDB	MongoDB データベースからデータを取得したり、MongoDB データベースでデータを更新したりします。
MQTT	MQTT (MQ Telemetry Transport) ブローカーからメッセージを取得したり、MQTT ブローカーへメッセージをパブリッシュしたりします。
OData	OData サービスからエンティティを取得したり、OData サービスによって管理されるエンティティを更新、作成、または削除します。
REST API	OpenAPI ドキュメントをアップロードして、カスタム REST API クライアントコネクタを作成します。その後、その REST API への接続を作成します。 インテグレーションの実行をトリガーする操作を定義する OpenAPI ドキュメントをアップロードして、REST API プロバイダーインテグレーションを作成します。Fuse Online でのアプリケーションの統合の REST API 呼び出しによってトリガーされるインテグレーションの作成 を参照してください。
Salesforce	Salesforce レコードを作成、更新、取得、または削除します。
SAP Concur	さまざまな SAP Concur アクションの1つを実行します。
ServiceNow	ServiceNow インスタンスからレコードを取得したり、ServiceNow インスタンスへレコードをコピーしたりします。
Slack	チャンネルからメッセージを取得したり、Slack チャンネルまたはユーザーにメッセージを送信したりします。

名前	説明
SQL データベース	Apache Derby、MySQL、または PostgreSQL データベースで SQL ステートメントまたは SQL ストアドプロシージャを呼び出します。他のタイプの SQL データベースに接続するには、そのデータベースの JDBC ドライバーが含まれる Fuse Online ライブラリーエクステンションをアップロードします。
Telegram	Telegram チャットボットを使用して、チャットからメッセージを取得したり、チャットにメッセージを送信したりします。
Twitter	ユーザーをメンションするツイートまたは指定のデータが含まれるツイートによって、シンプルなインテグレーションの実行をトリガーします。
Webhook	HTTP の GET または POST リクエストによってシンプルなインテグレーションの実行をトリガーします。

Fuse Online が必要なコネクタを提供しない場合、経験のある開発者であれば、カスタムコネクタを定義するエクステンションを作成して対応することができます。Fuse Online にアップロードする、エクステンションのコーディングやその **.jar** ファイルの作成に関する詳細は、以下を参照してください。

- [Tooling User Guide の Fuse ツーリングを使用したエクステンションの開発](#)
- [Fuse Online でのアプリケーションの統合のエクステンションのコーディング方法](#)

第2章 AMAZON WEB SERVICES への接続

Fuse Online インテグレーションは以下の Amazon Web サービスに接続できます。

- Simple Notification Service (SNS)
- Simple Queue Service (SQS)
- Simple Storage Service (S3)
- DynamoDB

Amazon Web サービスへのコネクションを作成するには、AWS クレデンシャルを取得する必要があります。詳細は、以下を参照してください。

- [「Fuse Online コネクションを作成するための AWS クレデンシャルの取得」](#)
- [「Amazon Simple Notification Service \(SNS\) への接続」](#)
- [「Amazon Simple Queue Service \(SQS\) への接続」](#)
- [「Amazon Simple Storage Service \(S3\) への接続」](#)
- [「Amazon DynamoDB への接続」](#)

2.1. FUSE ONLINE コネクションを作成するための AWS クレデンシャルの取得

Amazon サービスへのコネクションを作成するには、AWS アカウントに関連するアクセスキーが必要です。これは、Fuse Online コネクションがアクセスするリソースを作成したまたはこれから作成する AWS アカウントが該当します。リソースは S3 バケット、SNS トピック、または SQS キューになります。

AWS アクセスキーは、2つの部分で設定されます。

- Amazon アクセスキー ID
- Amazon シークレットアクセスキー

AWS はアクセスキーを使用して、Fuse Online コネクションからのリクエストを認証します。アクセスキーの作成や管理に関する AWS ドキュメントは多くあります。すでに AWS IAM (Identity and Access Management) ユーザーを作成し、そのユーザーに関連するアクセスキーをキャプチャーした場合、これらの値を指定し、S3、SNS、または SQS への Fuse Online コネクションを作成することができます。

アクセスキーを取得する必要がある場合は、以下の手順で取得できます。

前提条件

- コネクションがアクセスするリソースを作成した、AWS アカウントのログインクレデンシャルが必要です。
- または、新しいリソースを作成するためにコネクションが使用する AWS アカウントのログインクレデンシャルが必要です。

手順

1. <https://aws.amazon.com/> にアクセスし、コンソールにログインします。
2. コンソール右上のユーザー名の横にある下矢印をクリックし、**セキュリティ認証情報** をクリックします。
3. ポップアップダイアログの **Get Started with IAM Users** をクリックします。
このボタンのあるポップアップが表示されない場合は、左側の **ユーザー** をクリックします。
4. ユーザーを追加します。
 - a. **ユーザーを追加** をクリックします。
 - b. ユーザー名を入力し、**プログラムによるアクセス** を選択します。
 - c. **次のステップ: アクセス権限** をクリックします。
 - d. 新規ユーザーをグループに追加するプロンプトに応答します。
 - e. **次のステップ: タグ** をクリックします。
 - f. タグの追加は行わず、**次のステップ: 確認** をクリックします。
 - g. **ユーザーの追加** をクリックします。
5. **.csv のダウンロード** をクリックして、アクセスキーのローカルコピーを取得します。ダウンロードしたファイルには、Amazon サービスへの接続の作成に指定する必要がある、アクセスキー ID とシークレットアクセスキーが含まれます。
6. **Close** をクリックします。

その他のリソース

- [IAM ユーザーのアクセスキーの管理](#)
- [IAM のベストプラクティス](#)

2.2. AMAZON SIMPLE NOTIFICATION SERVICE (SNS) への接続

インテグレーションはメッセージを AWS SNS (Simple Notification Service) トピックへ送信できます。AWS SNS コネクションを作成し、そのコネクションをインテグレーションフローに追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。詳細は、以下を参照してください。

- [「Amazon SNS コネクションの作成」](#)
- [「Amazon SNS へのメッセージの送信」](#)

2.2.1. Amazon SNS コネクションの作成

Amazon SNS コネクションをインテグレーションに追加するには、Amazon SNS コネクションを作成する必要があります。

前提条件

- AWS アクセスキーが必要です。 [AWS クレデンシャルの取得](#) を参照してください。
- SNS トピックが存在するリージョンを知っている必要があります。このトピックが、作成する SNS コネクションのメッセージの送信先となります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクタを表示します。
3. **Amazon SNS** コネクタをクリックします。
4. **Access Key** フィールドに、Amazon アクセスキー ID を入力します。これは、コネクションのメッセージ送信先となるトピックを管理する、AWS アカウントのユーザーアクセスキーの一部です。
コネクションがアクセスするトピックが存在しない場合は、Fuse Online がインテグレーションの実行を開始しようとするときに、そのアクセスキーに関連する AWS アカウントを使用してトピックを作成します。しかし、他の AWS アカウントにトピックが存在する場合は、コネクションはトピックを作成できず、インテグレーションを開始できません。
5. **Secret Key** フィールドに、指定した Amazon アクセスキー ID の Amazon シークレットアクセスキーを入力します。
6. **Region** フィールドに、トピックが存在する AWS リージョンを選択します。コネクションがトピックを作成する場合は、選択したリージョンにトピックが作成されます。
7. **Validate** をクリックします。Fuse Online は即座にコネクションの検証を試み、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
8. 検証に成功したら **Next** をクリックします。
9. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**SNS North** を入力します。
10. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample SNS connection that sends messages to our north region topic.** を入力します。
11. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **SNS North** が表示されます。

2.2.2. Amazon SNS へのメッセージの送信

Amazon SNS トピックへメッセージを送信するには、Amazon SNS コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Amazon SNS コネクションが作成済みである必要があります。
- Fuse Online はインテグレーションの追加を要求したり、最後のコネクションをシンプルなインテグレーションへ追加するよう要求します。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. メッセージの送信に使用する Amazon SNS コネクションをクリックします。
3. **Send Object** アクションをクリックします。
4. **Topic Name or Amazon Resource Name** フィールドに、メッセージを送信する SNS トピックの名前またはトピックの Amazon Resource Name (ARN) を入力します。
SNS トピックが、コネクションのアクセスが許可される AWS アカウントに存在しなかったり、他の AWS アカウントに存在しない場合は、コネクションによるアクセスが設定されている AWS アカウントにトピックを作成します。
5. **Next** をクリックしてコネクションをインテグレーションに追加します。

結果

インテグレーションフローで追加した場所にコネクションが表示されます。

2.3. AMAZON SIMPLE QUEUE SERVICE (SQS) への接続

インテグレーションは、Amazon SQS キューからメッセージを取得したり、Amazon SQS キューにメッセージを送信したりすることができます。これには、Amazon SQS コネクションを作成し、そのコネクションをインテグレーションフローに追加します。詳細は、以下を参照してください。

- [「Amazon SQS コネクションの作成」](#)
- [「Amazon SQS からメッセージを取得してインテグレーションの実行をトリガー」](#)
- [「Amazon SQS キューへのメッセージの送信」](#)

2.3.1. Amazon SQS コネクションの作成

Amazon SQS コネクションをインテグレーションに追加するには、Amazon SQS コネクションを作成する必要があります。

前提条件

- AWS アクセスキーが必要です。[AWS クレデンシャルの取得](#) を参照してください。
- SQS キューが存在するリージョンを知っている必要があります。作成する SQS コネクションは、このキューに対してメッセージを送受信します。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクタを表示します。
3. **Amazon SQS** コネクタをクリックします。

4. **Access Key** フィールドに、Amazon アクセスキー ID を入力します。これは、コネクションがアクセスするキューを管理する AWS アカウントのユーザーアクセスキーの一部です。コネクションがアクセスするキューが存在しない場合は、Fuse Online がインテグレーションの実行を開始しようとするときに、そのアクセスキーに関連する AWS アカウントを使用してキューを作成します。しかし、他の AWS アカウントにキューが存在する場合は、コネクションはキューを作成できず、インテグレーションを開始できません。
5. **Secret Key** フィールドに、指定した Amazon アクセスキー ID の Amazon シークレットアクセスキーを入力します。
6. **Region** フィールドに、キューが存在する AWS リージョンを選択します。コネクションがキューを作成する場合は、作成したリージョンにキューが作成されます。
7. **Validate** をクリックします。Fuse Online は即座にコネクションの検証を試み、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
8. 検証に成功したら **Next** をクリックします。
9. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**SQS South** を入力します。
10. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample SQS connection that sends messages to our southern region queue.** を入力します。
11. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **SQS South** が表示されます。

2.3.2. Amazon SQS からメッセージを取得してインテグレーションの実行をトリガー

Amazon SQS キューからメッセージを取得してインテグレーションを開始するには、Amazon SQS コネクションをシンプルなインテグレーションの最初のコネクションとして追加します。

前提条件

- メッセージの取得元のキューにアクセスするよう設定されている Amazon SQS コネクションが作成済みである必要があります。
- コネクションがメッセージを取得する SQS キューの名前を知っている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する Amazon SQS コネクションをクリックします。
4. **Choose an action** ページで、**Poll an Amazon SQS Queue** を選択し、SQS キューからメッセージを定期的に取得します。
5. このアクションを設定するには以下を行います。

- a. **Delay** フィールドで、デフォルトの 500 ミリ秒をポーリング間の経過時間として受け入れます。別のポーリング間隔を指定する場合は、数値を入力し、時間の単位を選択します。
- b. **Maximum Objects to Retrieve** フィールドに、1 回のポーリング操作で取得可能なメッセージの最大数を入力します。デフォルトは 10 です。
取得するメッセージの数を無制限にするには、**0** または負の整数を指定します。**Maximum Objects to Retrieve** が無制限の場合、ポーリングは SQS キューのすべてのメッセージを取得します。
- c. **Queue Name or Amazon Resource Name** フィールドに、メッセージを取得する SQS キューの名前またはキューの ARN を指定します。この名前を持つキューが存在しない場合は、コネクションがこの名前で作成します。必須のフィールドはこのフィールドのみです。
- d. **Obtain messages and then delete them from the queue**(メッセージを取得した後にキューから削除) を指定するかどうかを示します。メッセージを取得した後にキューから削除するのはデフォルトの動作です。メッセージを取得し、キューにそのメッセージを残す場合のみ、このオプションの選択を解除します。キューに取得済みのメッセージを残すと、再度取得することになります。
- e. **Obtain messages and delete the message if it does make it through a Camel filter**(メッセージを取得し、Camel フィルターを通過した場合はメッセージを削除) を指定するかどうかを示します。このオプションはデフォルトで選択されます。取得したメッセージをフィルターに通し、キューから削除するかどうかを判断する場合はデフォルトのままにします。**Obtain messages and then delete from the queue**の選択を解除する必要はありません。**Obtain messages and delete the message if it does make it through a Camel filter** を選択すると、Fuse Online によって正しい動作が実行されます。

6. **Next** をクリックします。

結果

インテグレーションフローの最初にコネクションが表示されます。

次のステップ

Obtain messages and delete the message if it does make it through a Camel filterを選択した場合、この SQS コネクションの後にフィルターステップを追加する必要があります。このフィルターステップは SQS コネクションの直後に追加する必要はありません。すべてのコネクションをインテグレーションに追加した後、フィルターステップを追加します。実行中にフィルターを通過したメッセージがあると、Fuse Online は同じコネクションである SQS の最初のコネクションを使用して、これらのメッセージをキューから削除します。

2.3.3. Amazon SQS キューへのメッセージの送信

フローの途中でメッセージを Amazon SQS キューに送信したり、メッセージを Amazon SQS キューに送信してシンプルなインテグレーションを終了することができます。

前提条件

- メッセージの送信先のキューにアクセスするよう設定されている Amazon SQS コネクションが作成済みである必要があります。
- コネクションがメッセージを送信する SQS キューの名前を知っている必要があります。
- Fuse Online はインテグレーションの追加を要求したり、シンプルなインテグレーションの最後のコネクションを選択するよう要求します。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. メッセージの送信に使用する Amazon SQS コネクションをクリックします。
3. コネクションが実行するアクションを選択します。
 - **Send Object** を選択すると、1つのメッセージをキューに送信します。
 - **Send Batch Object** を選択すると、メッセージのバッチをキューに送信します。
4. FIFO (先入れ先出し) キューの場合は、**Message group ID strategy** フィールドで **ConstantMessageGroupIdStrategy** を使用するか、フィールドをクリックして別のストラテジーを選択します。

シンプルなキューでは、このパラメーターを設定しても影響はありません。メッセージグループ ID は、メッセージが属するグループを識別します。同じメッセージグループに属するメッセージは、常にメッセージグループの厳密な順序で1つずつ処理されます。メッセージグループ ID ストラテジーは、コネクションがグループ ID をメッセージに割り当てる方法を判断します。オプションは次のとおりです。

 - **ConstantMessageGroupIdStrategy**: コネクションは定数を使用してメッセージをグループ化します。
 - **ExchangeIdMessageGroupIdStrategy**: コネクションは、各メッセージのエクステンジ ID を使用してメッセージをグループ化します。
 - **PropertyValueMessageGroupIdStrategy**: コネクションは内部プロパティの値を使用してメッセージをグループ化します。

FIFO キューは、メッセージが送受信される順番が厳密に保持され、各メッセージが1度だけ処理されるように設計されています。FIFO キューの名前には、**.fifo** が接尾辞として付きます。
5. **Message deduplication ID strategy** フィールドの FIFO キューに **NullMessageDeduplicationIdStrategy** を使用するか、フィールドをクリックして **ExchangeIdMessageDeduplicationIdStrategy** を選択します。

シンプルなキューでは、このパラメーターを設定しても影響はありません。FIFO キューでは、コネクションは選択したストラテジーを使用して、重複したメッセージをキューに送信しないようにします。

 - **NullMessageDeduplicationIdStrategy** はメッセージのボディー部を重複排除要素として使用します。つまり、コネクションはメッセージのボディー部を比較して、重複メッセージを特定します。
 - **ExchangeIdMessageDeduplicationIdStrategy** はメッセージのエクステンジ ID を重複排除要素として使用します。つまり、コネクションはエクステンジ ID を比較して重複メッセージを特定します。
6. **Queue Name or Amazon Resource Name** フィールドに、メッセージを送信するキューの名前を入力します。キューが存在しない場合は、コネクションが作成します。
7. **Delay** フィールドで、0 (遅延なし) のデフォルトを使用するか、数字を指定して遅延を追加します。または、遅延を追加するには数字を指定します。インテグレーションは、この秒数後、メッセージを送信します。これは、メッセージコンシューマーがコネクションが送信するメッセージにすぐに対応できない場合に便利です。

8. **Next** をクリックします。

結果

インテグレーションフローで追加した場所に接続が表示されます。

その他のリソース

- [Amazon SQS メッセージ重複排除 ID の使用](#)
- [Amazon SQS メッセージグループ ID の使用](#)

2.4. AMAZON SIMPLE STORAGE SERVICE (S3) への接続

インテグレーションは Amazon S3 バケットからデータを取得したり、Amazon S3 バケットへデータをコピーしたりすることができます。これには、Amazon S3 接続を作成し、その接続をインテグレーションフローに追加します。詳細は、以下を参照してください。

- [「Amazon S3 接続の作成」](#)
- [「Amazon S3 からデータを取得してインテグレーションの実行をトリガー」](#)
- [「Amazon S3 にデータを追加してインテグレーションを終了」](#)
- [「インテグレーションの途中で Amazon S3 にデータを追加」](#)

2.4.1. Amazon S3 接続の作成

Amazon S3 接続をインテグレーションに追加する前に Amazon S3 接続を作成する必要があります。

前提条件

- AWS アクセスキーが必要です。[AWS クレデンシャルの取得](#) を参照してください。
- 接続がアクセスするバケットがすでに存在する場合、以下を知っている必要があります。
 - バケットのあるリージョン。
 - バケットの名前または Amazon Resource Name (ARN)。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能な接続を表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクタを表示します。
3. **Amazon S3** コネクタをクリックします。
4. **Access Key** フィールドに、Amazon アクセスキー ID を入力します。これは、接続がアクセスするバケットを管理する AWS アカウントのユーザーアクセスキーの一部です。接続がアクセスするバケットが存在しない場合は、Fuse Online がインテグレーションの実行を開始しようとするときに、そのアクセスキーに関連する AWS アカウントを使用してバケットを作成します。しかし、他の AWS アカウントにバケットが存在する場合は、コネク

シオンはバケットを作成できず、インテグレーションを開始できません。

5. **Secret Key** フィールドに、指定した Amazon アクセスキー ID の Amazon シークレットアクセスキーを入力します。
6. **Region** フィールドではバケットが存在する AWS リージョンを選択します。接続がバケットを作成する場合にここで選択されたリージョンに作成されます。
7. **Bucket Name or Amazon Resource Name** フィールドに、この接続がアクセスするバケットの名前またはバケットの ARN を入力します。
指定したバケットが存在しない場合、接続は指定した名前でバケットを作成しようとします。S3 では、バケットをパブリックにアクセスできる URL として使用することが可能であるため、指定するバケット名はグローバルで一意である必要があります。さらに、[S3 バケットの命名要件](#) に準拠する必要があります。
8. **Validate** をクリックします。Fuse Online は即座に接続の検証を試み、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
9. 検証に成功したら **Next** をクリックします。
10. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**Obtain S3 Data** を入力します。
11. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。たとえば、**Sample S3 connection that obtains data from a northeast bucket.** を入力します。
12. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **Obtain S3 Data** が表示されます。

2.4.2. Amazon S3 からデータを取得してインテグレーションの実行をトリガー

Amazon S3 バケットからデータを取得してインテグレーションを開始するには、Amazon S3 接続をシンプルなインテグレーションの最初の接続として追加します。

前提条件

Amazon S3 接続が作成済みである必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する Amazon S3 接続をクリックします。
4. **Choose an action** ページで、接続が実行するアクションを選択します。
 - **Get Object** は、接続がアクセスするバケットからファイルを取得します。**File Name** フィールドでは、取得するファイルの名前を入力します。指定したファイルがバケットにない場合は、ランタイムエラーが発生します。
 - **Poll an Amazon S3 Bucket** は、接続がアクセスするバケットからファイルを定期的に取得します。このアクションを設定するには、以下を行います。

- a. **Delay** フィールドで、デフォルトの 500 ミリ秒をポーリング間の経過時間として受け入れます。別のポーリング間隔を指定する場合は、数値を入力し、時間の単位を選択します。
 - b. **Maximum Objects to Retrieve** フィールドに、1回のポーリング操作で取得可能なファイルの最大数を入力します。デフォルトは 10 です。
取得するファイルの数を無制限にするには、**0** または負の整数を指定します。**Maximum Objects to Retrieve** が無制限の場合、ポーリングはバケットのすべてのファイルを取得します。

バケットに指定された最大数を超える数のファイルが含まれている場合、編集または作成日時が最も新しいファイルから取得します。
 - c. **Prefix** フィールドには、文字列を評価する正規表現を任意で指定します。接頭辞を指定した場合、名前がこの文字列で始まるファイルのみを取得します。
 - d. **Obtain files and then delete them from the bucket**(ファイルを取得し、取得済みのファイルをバケットから削除)を行うかどうかを示します。
5. アクションの設定後、**Next** をクリックしてアクションの出力タイプを指定します。
 6. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
 7. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 8. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 9. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 10. **Next** をクリックします。

結果

インテグレーションに最初のコネクションが追加され、Fuse Online は最後のコネクションを選択するよう要求します。

2.4.3. Amazon S3 にデータを追加してインテグレーションを終了

データを Amazon S3 にコピーしてインテグレーションを終了するには、Amazon S3 コネクションをシンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Amazon S3 コネクションが作成済みである必要があります。
- シンプルなインテグレーションを作成または編集することになります。Fuse Online は最後のコネクションを選択するよう要求します。

手順

1. **Choose a connection** ページで、インテグレーションの終了に使用する Amazon S3 コネクションをクリックします。
2. コネクションが実行するアクションを選択します。
 - **Copy Object** は1つ以上のオブジェクトをバケットに追加します。1つのファイルをバケットに追加するには、その名前を **File Name** フィールドに入力します。

複数のファイルをバケットに追加する場合はファイル名を指定しません。この場合、アクションによって以前のインテグレーションステップから取得されるすべてのオブジェクトが追加されます。

ポーリングを使用して複数のファイルを取得し、ファイル名を指定する場合、**Copy Object** アクションはポーリングで受け取った最後のファイルのみを追加します。
 - **Delete Object** はバケットからオブジェクトを削除します。**File Name** フィールドに削除するオブジェクトの名前を指定します。指定したファイルがバケットにない場合、エラーは発生せず、インテグレーションが継続されます。
3. 指定したアクションの設定後、**Next** をクリックしてアクションの入力タイプを指定します。
4. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。

5. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、JSON schema を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
6. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
7. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
8. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションの最後にコネクションが表示されます。

2.4.4. インテグレーションの途中で Amazon S3 にデータを追加

インテグレーションの途中で Amazon S3 にデータを追加するには、Amazon S3 コネクションをフローの途中で追加します。

前提条件

- Amazon S3 コネクションが作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。

手順

1. フロービジュアライゼーションで、Amazon S3 コネクションを追加する場所のプラス記号をクリックします。
2. フローの途中で使用する Amazon S3 コネクションをクリックします。
3. コネクションが実行するアクションを選択します。
 - **Copy Object** は1つ以上のオブジェクトをバケットに追加します。
1つのファイルをバケットに追加するには、その名前を **File Name** フィールドに入力します。

複数のファイルをバケットに追加する場合はファイル名を指定しません。この場合、アクションによって以前のインテグレーションステップから取得されるすべてのオブジェクトが追加されます。

ポーリングを使用して複数のファイルを取得し、ファイル名を指定する場合、**Copy Object** アクションはポーリングで受け取った最後のファイルのみを追加します。

 - **Delete Object** はバケットからオブジェクトを削除します。**File Name** フィールドに削除するオブジェクトの名前を指定します。指定したファイルがバケットにない場合、エラーは発生せず、インテグレーションが継続されます。

4. 指定したアクションの設定後、**Next** をクリックしてアクションの入力タイプを指定します。
5. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
6. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
7. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
8. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
9. **Next** をクリックします。

結果

フロービジュアライゼーションで追加した場所にコネクションが表示されます。

2.5. AMAZON DYNAMODB への接続

インテグレーションは、Amazon DynamoDB テーブルからデータを取得したり、DynamoDB テーブルへデータを追加したりでき、DynamoDB テーブルからデータを削除することもできます。これには、Amazon DynamoDB コネクションを作成し、そのコネクションをインテグレーションフローに追加します。



重要

DynamoDB への接続はテクノロジープレビューの機能です。テクノロジープレビュー機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、Red Hat は本番環境での使用は推奨しません。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能により、近日発表予定の製品機能をリリースに先駆けてご提供でき、お客様は開発プロセス時に機能をテストして、フィードバックをお寄せいただくことができます。Red Hat のテクノロジープレビュー機能のサポート範囲に関する詳細は、<https://access.redhat.com/ja/support/offerings/techpreview> を参照してください。

詳細は、以下を参照してください。

- [「Amazon DynamoDB コネクションの作成」](#)
- [「Amazon DynamoDB テーブルからデータを取得」](#)
- [「Amazon DynamoDB テーブルへのデータの追加」](#)
- [「Amazon DynamoDB テーブルからデータを削除」](#)

2.5.1. Amazon DynamoDB コネクションの作成

インテグレーションの Amazon DynamoDB へ接続する前に、Amazon DynamoDB コネクションを作成する必要があります。

前提条件

- AWS アクセスキーが必要です。[AWS クレデンシャルの取得](#) を参照してください。
- コネクションがアクセスする DynamoDB テーブルが含まれる AWS リージョンを知っている必要があります。
- コネクションがアクセスする DynamoDB テーブルの名前を知っている必要があります。このテーブルは、インテグレーションが DynamoDB に接続するときに存在している必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクタを表示します。
3. **AWS DDB コネクタ** をクリックします。
4. **Access Key** フィールドに、Amazon アクセスキー ID を入力します。これは、コネクションがアクセスする DynamoDB テーブルを管理する AWS アカウントのユーザーアクセスキーの一部です。
5. **Secret Key** フィールドに、指定した Amazon アクセスキー ID の Amazon シークレットアクセスキーを入力します。
6. **Region** フィールドで、DynamoDB テーブルが存在する AWS リージョンを選択します。

7. **Table Name** フィールドに、この接続がアクセスする DynamoDB テーブルの名前を入力します。
8. **Validate** をクリックします。Fuse Online は即座に接続の検証を試み、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
9. 検証に成功したら **Next** をクリックします。
10. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**DynamoDB West** を入力します。
11. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。たとえば、**Sample DynamoDB connection that accesses sales data in the western region.** を入力します。
12. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **DynamoDB West** が表示されます。

結果

Fuse Online がインテグレーションの追加を要求すると、直前に作成された DynamoDB 接続が表示されます。この接続は、常に接続の作成時に指定した DynamoDB テーブルのみにアクセスします。

2.5.2. Amazon DynamoDB テーブルからデータを取得

インテグレーションフローで、Amazon DynamoDB テーブルからアイテムを取得するには、Amazon DynamoDB 接続をフローの途中に追加します。

DynamoDB 接続は、シンプルなインテグレーションを開始できません。DynamoDB テーブルから定期的にデータを取得する場合は、タイマーを使ってシンプルなインテグレーションを開始した後、テーブルをクエリーする DynamoDB 接続を開始します。

前提条件

- Amazon DynamoDB 接続が作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。

手順

1. フロービジュアライゼーションで、Amazon DynamoDB 接続を追加する場所のプラス記号をクリックします。
2. **Choose a connection** ページで、データの取得元となるテーブルにアクセスするよう設定された Amazon DynamoDB 接続をクリックします。
3. **Choose an action** ページで **Query** アクションを選択します。
4. **Filter** フィールドで、接続が返すアイテムのキー属性を識別する JSON 表記を指定します。:# を使って入力パラメーターを指定できます。たとえば、2つのキー属性があるアイテムを取得する場合、以下のように指定することが可能です。

```
 {"key1":":#PARAM1", "key2":":#PARAM2"}
```

- 任意手順: **Attributes to query** フィールドに、コネクションが返すデータを含む属性名のコンマ区切りリストを入力します。以下に例を示します。

```
 key1,key2,attribute3,attribute4,attribute5
```

属性を指定しないと、クエリーはキー属性を返します。

- Next** をクリックします。

結果

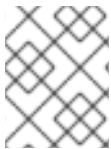
これで、インテグレーションにデータを取得する DynamoDB コネクションが含まれます。起動時に、コネクションはクエリーの結果が含まれる JSON インスタンスドキュメントを返します。

次のステップ

Filter フィールドにプレースホルダーパラメーターを指定した場合、このコネクションの前にデータマッピングステップを追加します。このデータマッピングステップで、ソースフィールドを DynamoDB コネクションのターゲットプレースホルダーフィールドにマップします。

2.5.3. Amazon DynamoDB テーブルへのデータの追加

DynamoDB コネクションは、フローの途中でアイテムを DynamoDB テーブルに追加したり、シンプルなインテグレーションを終了するためにアイテムを DynamoDB テーブルに追加したりすることができます。これには、DynamoDB コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。



注記

本リリースでは、DynamoDB コネクションは DynamoDB テーブルのアイテムを更新できません。これは今後のリリースで変更される予定です。

前提条件

- Amazon DynamoDB コネクションが作成済みである必要があります。
- シンプルなインテグレーションを作成または編集することになり、Fuse Online はインテグレーションへの追加または最後のコネクションの選択を要求します。

手順

- Choose a connection** ページで、追加先となるテーブルにアクセスするよう設定された Amazon DynamoDB コネクションをクリックします。
- Choose an action** ページで **Put Item** を選択します。
- JSON definition of the element** フィールドで、追加されるテーブルアイテムを定義する JSON 表記を入力します。アイテムのキー属性を指定する必要があります。:# で始まる入力パラメータープレースホルダーが許可されます。保存するすべての属性と値を指定してください。以下の例では、**userID** キーになります。

```
 {"userID": "aslan", "attr1": "some-value", "attr2": "another-value"}
```

4. **Next** をクリックします。

結果

コネクションは、インテグレーションビジュアライゼーションで追加した場所に表示されます。実行中、コネクションがアクセスするように設定された DynamoDB テーブルに、定義された JSON アイテムを追加します。

次のステップ

アイテムの JSON 定義にプレースホルダーパラメーターを指定した場合、このコネクションの前にデータマッピングステップを追加します。このデータマッピングステップで、ソースフィールドを DynamoDB コネクションのターゲットプレースホルダーフィールドにマップします。

2.5.4. Amazon DynamoDB テーブルからデータを削除

DynamoDB コネクションは、フローの途中でアイテムを DynamoDB テーブルから削除したり、シンプルなインテグレーションを終了するためにアイテムを DynamoDB テーブルに追加したりすることができます。これには、DynamoDB コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Amazon DynamoDB コネクションが作成済みである必要があります。
- シンプルなインテグレーションを作成または編集することになり、Fuse Online はインテグレーションへの追加または最後のコネクションの選択を要求します。

手順

1. **Choose a connection** ページで、削除するアイテムが含まれるテーブルにアクセスするよう設定された Amazon DynamoDB コネクションをクリックします。
2. **Choose an action** ページで **Remove Item** を選択します。
3. **JSON definition of the element** フィールドで、削除するアイテムを定義する JSON 表記を入力します。アイテムのキー属性を指定する必要があります。:# で始まる入力パラメーターが許可されます。以下の例では、**userID** キーになります。

```
["userID":"aslan"]
```

4. **Next** をクリックします。

結果

フロービジュアライゼーションで追加した場所にコネクションが表示されます。

次のステップ

JSON filter of the element フィールドにプレースホルダーパラメーターを指定した場合、このコネクションの前にデータマッピングステップを追加します。このデータマッピングステップで、ソースフィールドを DynamoDB コネクションの各ターゲットプレースホルダーフィールドにマップします。

第3章 AMQ への接続

インテグレーションでは、Red Hat AMQ ブローカーからメッセージを取得したり、Red Hat AMQ ブローカーへメッセージをパブリッシュすることができます。Red Hat AMQ は、クライアントとメッセージブローカーとの間の通信に OpenWire プロトコルを使用します。以下のブローカータイプと通信するには、Red Hat AMQ コネクタを使用して、対象のブローカーへの接続を作成します。

- AMQP をサポートしない Apache ActiveMQ ブローカー
- AMQ 6 ブローカー

以下のブローカータイプの1つと通信するには、[AMQP コネクタを使用](#)して、対象のブローカーへの接続を作成します。

- AMQP をサポートする Apache ActiveMQ ブローカー
- Apache ActiveMQ Artemis
- AMQ 7 ブローカー
- EnMasse (オープンソースのメッセージングプラットフォーム)

Red Hat AMQ コネクタを使用するには、以下を参照してください。

- [「AMQ 接続の作成」](#)
- [「AMQ 接続を追加してメッセージ受信時にインテグレーションの実行をトリガー」](#)
- [「AMQ メッセージをパブリッシュしてインテグレーションを終了」](#)
- [「インテグレーションの途中で AMQ メッセージを公開」](#)

3.1. AMQ 接続の作成

インテグレーションでは、以下からメッセージを取得したり、以下にメッセージをパブリッシュすることができます。

- AMQP をサポートしない Apache ActiveMQ ブローカー
- AMQ 6 ブローカー

このような場合、インテグレーションに追加できる Red Hat AMQ 接続を作成します。

前提条件

接続する Red Hat AMQ ブローカーには以下が必要となります。

- ブローカーの URL
- ユーザーアカウントのクレデンシャル
- ブローカーの PEM 形式の証明書

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能な接続を表示します。

2. 右上の **Create Connection** をクリックしてコネクタを表示します。
3. **Red Hat AMQ** コネクタを表示します。
4. コネクションを設定します。
 - a. **Broker URL** フィールドにデータの送信先またはデータの取得元の場所を入力します (例: **tcp://localhost:61616**)。
 - b. **User name** フィールドに、このブローカーへアクセスするために使用するアカウントのユーザー名を入力します。
 - c. **Password** フィールドに、このブローカーへアクセスするために使用するアカウントのパスワードを入力します。
 - d. **Client ID** フィールドに、メッセージを見逃さずにコネクションを開閉できるようにする ID を入力します。宛先のタイプはトピックにする必要があります。
 - e. このコネクションが開発環境で使用される場合は、**Check certificates** を無効にすると時間を一部節約することができます。証明書確認の無効化は開発環境で便利です。セキュアな本番環境では、常に **Check certificates** を有効にするようにしてください。
 - f. **Broker certificate** フィールドに Red Hat AMQ ブローカーの PEM 形式の証明書テキストを貼り付けます。これは、証明書確認を無効にした場合以外は必須です。
 - g. **Client certificate** フィールドに Red Hat クライアントの PEM 形式の証明書テキストを貼り付けます。このフィールドは常に任意です。
5. **Validate** をクリックします。Fuse Online は即座にコネクションを検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
6. 検証に成功した場合は **Next** をクリックします。
7. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Red Hat AMQ 1** を入力します。
8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample Red Hat AMQ connection that uses a provided broker.** を入力します。
9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Red Hat AMQ 1** が表示されます。

3.2. AMQ コネクションを追加してメッセージ受信時にインテグレーションの実行をトリガー

Red Hat AMQ ブローカーからメッセージを受信したときにインテグレーションの実行をトリガーするには、Red Hat AMQ コネクションを最初のコネクションとして追加します。

前提条件

メッセージの取得元となる Red Hat AMQ ブローカーへの Red Hat AMQ コネクションが作成されている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する Red Hat AMQ コネクションをクリックします。
4. **Choose an action** ページで、**Subscribe for messages** アクションを選択し、指定したキューまたはトピックからメッセージを受信します。
5. アクションを設定するには以下を行います。
 - a. **Destination name** フィールドに、データの受信元となるキューまたはトピックの名前を入力します。
 - b. **Destination type** では **Queue** を許可するか、**Topic** を選択します。
 - c. メッセージを見逃さずにコネクションを開閉できるようにするため、**Durable subscription ID** フィールドに永続サブスクリプション ID を入力します。宛先のタイプはトピックにする必要があります。
 - d. 特定の条件を満たすデータのみを受信する場合は、**Message selector** フィールドにフィルター式を入力します。
 メッセージセレクターは式が含まれる文字列です。式の構文は、SQL92 条件式構文のサブセットを基にしています。以下の例のメッセージセレクターは、値が **Sports** または **Opinion** に設定された **NewsType** プロパティがあるメッセージをすべて選択します。
NewsType = 'Sports' OR NewsType = 'Opinion'
 メッセージコンシューマーは、ヘッダーとプロパティがメッセージセレクターの式と一致するメッセージのみを受信します。メッセージセレクターは、メッセージのボディー部の内容を基にしてメッセージを選択することはできません。
6. **Next** をクリックして、アクションの出力タイプを指定します。
7. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
8. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。

○ **Destination Name** フィールドには、データのタイプに選択した名前を入力します。たとえば、**Queue**

9. **Data Type Name** ノールトにはアータタイプに選択した名前を入力します。たとえば、ヘンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときはこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
10. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
11. **Next** をクリックします。

結果

インテグレーションフローの最初にコネクションが表示されます。

3.3. AMQ メッセージをパブリッシュしてインテグレーションを終了

メッセージを Red Hat AMQ ブローカーにパブリッシュしてシンプルなインテグレーションを終了するには、Red Hat AMQ コネクションをシンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- メッセージの公開先となる Red Hat AMQ ブローカーへのコネクションが作成されている。
- インテグレーションを作成することになり、Fuse Online は最後のコネクションの選択を要求します。または、インテグレーションを変更して最後のコネクションを変更することになります。

手順

1. **Choose a connection** ページで、インテグレーションの終了に使用する Red Hat AMQ コネクションをクリックします。
2. **Choose an action** ページで、**Publish messages** アクションを選択し、指定したキューまたはトピックにメッセージを公開します。
3. **Destination name** フィールドに、メッセージの送信先となるキューまたはトピックの名前を入力します。
4. **Destination type** では **Queue** を許可するか、**Topic** を選択します。
5. **Persistent** を選択し、コネクションが失敗した場合でもメッセージが確実に送信されるようにします。
6. **Next** をクリックし、アクションの入力および出力タイプを指定します。
7. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。
これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。

- **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
8. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 9. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 10. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 11. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションの最後にコネクションが表示されます。

3.4. インテグレーションの途中で AMQ メッセージを公開

インテグレーションの途中で Red Hat AMQ ブローカーにメッセージをパブリッシュするには、Red Hat AMQ コネクションをフローの途中で追加します。

前提条件

- メッセージの公開先となる Red Hat AMQ ブローカーへのコネクションが作成されている。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. フローの途中で追加する Red Hat AMQ コネクションをクリックします。
3. **Choose an action** ページで、以下のアクションの1つを選択します。
 - **Publish messages** は指定するキューまたはトピックにメッセージを公開します。このアクションを設定するには、以下を行います。
 - a. **Destination name** フィールドに、メッセージの送信先となるキューまたはトピックの名前を入力します。

- b. **Destination type** では **Queue** を許可するか、**Topic** を選択します。
 - c. **Persistent** を選択し、コネクションが失敗した場合でもメッセージが確実に送信されるようにします。
- **Request response using messages** で、応答の受信に指定した JMS 宛先にメッセージを送ります。このアクションを設定するには、以下を行います。
 - a. **Destination name** フィールドに、メッセージの送信先となるキューまたはトピックの名前を入力します。
 - b. **Destination type** では **Queue** を許可するか、**Topic** を選択します。
 - c. 特定の条件を満たす応答のみを受信する場合は、**Message selector** フィールドにフィルター式を入力します。
メッセージセレクターは式が含まれる文字列です。式の構文は、SQL92 条件式構文のサブセットを基にしています。以下の例のメッセージセレクターは、値が **Sports** または **Opinion** に設定された **NewsType** プロパティがあるメッセージをすべて選択します。

NewsType = 'Sports' OR NewsType = 'Opinion'

メッセージコンシューマーは、ヘッダーとプロパティがメッセージセレクターの式と一致するメッセージのみを受信します。メッセージセレクターは、メッセージのボディ部の内容を基にしてメッセージを選択することはできません。
 - d. **Named reply to** フィールドには、キューまたはトピックの名前を入力します。宛先によって、応答がこのキューまたはトピックに送信されます。
 - e. **Persistent** を選択し、コネクションが失敗した場合でもメッセージが確実に送信されるようにします。
 - f. **Response time out** フィールドには、ランタイム例外を出力する前にこのコネクションが応答メッセージを待つ期間 (ミリ秒単位) を指定します。デフォルトは 5000 ミリ秒 (5 秒) です。
4. **Next** をクリックし、アクションの入力タイプを指定してから、アクションの出力タイプを指定します。
 5. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。

6. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
7. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
8. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
9. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

第4章 AMQP への接続

インテグレーションでは、AMQP (Advanced Message Queue Protocol) ブローカーからメッセージを取得したり、AMQP ブローカーへメッセージをパブリッシュすることができます。AMQP は、クライアントとメッセージブローカーとの間の通信を定義します。以下のブローカータイプで通信するには、AMQP コネクタを使用して対象のブローカーへの接続を作成します。

- AMQP をサポートする Apache ActiveMQ ブローカー
- Apache ActiveMQ Artemis
- AMQ 7 ブローカー
- EnMasse (オープンソースのメッセージングプラットフォーム)

以下のブローカータイプの1つと通信するには、[Red Hat AMQ コネクタを使用](#) して、対象のブローカーへの接続を作成します。

- AMQP をサポートしない Apache ActiveMQ ブローカー
- AMQ 6 ブローカー



注記

AMQP コネクタを使用して AMQP をサポートしない Apache ActiveMQ ブローカーへの接続や AMQ 6 ブローカーへの接続を作成できます。これを行うには、ブローカーのトランスポートを設定する必要があります。ブローカーの設定に関する情報は、[Red Hat JBoss A-MQ Managing and Monitoring Brokers の Adding Client Connection Points](#) を参照してください。指定する設定値に関する情報は、[Red Hat JBoss A-MQ Connection Reference の Advanced Message Queuing Protocol \(AMQP\)](#) を参照してください。

AMQP コネクタを使用するには、以下を参照してください。

- [「AMQP 接続の作成」](#)
- [「AMQP 接続を追加してメッセージ受信時にインテグレーションの実行をトリガー」](#)
- [「フローの途中で AMQP へメッセージをパブリッシュまたは AMQP へメッセージをパブリッシュしてインテグレーションを終了」](#)

4.1. AMQP 接続の作成

インテグレーションで AMQP ブローカーからメッセージを取得したり AMQP ブローカーへメッセージをパブリッシュする場合は、インテグレーションに追加できる AMQP 接続を作成します。

前提条件

接続する AMQP ブローカーに以下が必要となります。

- URI
- ユーザーアカウントのクレデンシャル
- PEM 形式の証明書テキスト

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックしてコネクタを表示します。
3. **AMQP Message Broker** コネクタをクリックします。
4. コネクションを設定します。
 - a. **Connection URI** フィールドにデータの送信先または取得元となる場所を入力します。
 - b. **User name** フィールドに、このブローカーへアクセスするために使用するアカウントのユーザー名を入力します。
 - c. **Password** フィールドに、このブローカーへアクセスするために使用するアカウントのパスワードを入力します。
 - d. **Client ID** フィールドに、メッセージを見逃さずにコネクションを開閉できるようにする ID を入力します。宛先のタイプはトピックにする必要があります。
 - e. このコネクションが開発環境で使用される場合は、**Check certificates** を無効にすると時間を一部節約することができます。証明書確認の無効化は開発環境で便利です。セキュアな本番環境では、常に **Check certificates** を有効にするようにしてください。
 - f. **Broker certificate** フィールドにブローカーの PEM 形式の証明書テキストを貼り付けます。これは、証明書の確認を無効にした場合以外は必須になります。
 - g. **Client certificate** フィールドにクライアントの PEM 形式の証明書テキストを貼り付けます。このフィールドは常に任意です。
5. **Validate** をクリックします。Fuse Online は即座にコネクションを検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
6. 検証に成功した場合は **Next** をクリックします。
7. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**AMQP 1** を入力します。
8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample AMQP connection** を入力します。
9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **AMQP 1** が表示されます。

4.2. AMQP コネクションを追加してメッセージ受信時にインテグレーションの実行をトリガー

AMQP ブローカーからメッセージを受信したときにインテグレーションの実行をトリガーするには、AMQP コネクションをインテグレーションの最初のコネクションとして追加します。

前提条件

- メッセージの受信元となる AMQP ブローカーへの接続が作成されている必要があります。
- インテグレーションを作成することになります。Fuse Online は最初の接続の選択を要求します。

手順

1. **Choose a connection** ページで、インテグレーションを開始するために使用する AMQP 接続をクリックします。
2. **Choose an action** ページで、**Subscribe for messages** アクションを選択し、指定したキューまたはトピックからメッセージを受信します。
3. アクションを設定するには以下を行います。
 - a. **Destination name** フィールドに、データの受信元となるキューまたはトピックの名前を入力します。
 - b. **Destination type** では **Queue** を許可するか、**Topic** を選択します。
 - c. メッセージを見逃さずに接続を開閉できるようにするため、**Durable subscription ID** フィールドに永続サブスクリプション ID を入力します。宛先のタイプはトピックにする必要があります。
 - d. 特定の条件を満たすデータのみを受信する場合は、**Message selector** フィールドにフィルター式を入力します。
 メッセージセレクトターは式が含まれる文字列です。式の構文は、SQL92 条件式構文のサブセットを基にしています。以下の例のメッセージセレクトターは、値が **Sports** または **Opinion** に設定された **NewsType** プロパティがあるメッセージをすべて選択します。
NewsType = 'Sports' OR NewsType = 'Opinion'
 メッセージコンシューマーは、ヘッダーとプロパティがメッセージセレクトターの式と一致するメッセージのみを受信します。メッセージセレクトターは、メッセージのボディ部分の内容を基にしてメッセージを選択することはできません。
4. **Next** をクリックして、アクションの出力タイプを指定します。
5. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。

6. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
7. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
8. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
9. **Next** をクリックします。

結果

インテグレーションフローの最初にコネクションが表示されます。

4.3. フローの途中で AMQP へメッセージをパブリッシュまたは AMQP へメッセージをパブリッシュしてインテグレーションを終了

フローの途中で AMQP ブローカーへメッセージをパブリッシュしたり、AMQP ブローカーへメッセージをパブリッシュしてシンプルなインテグレーションを終了することができます。これには、AMQP コネクションをフローの途中で追加するか、AMQP コネクションをシンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- メッセージの公開先となる AMQP ブローカーへのコネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online でシンプルなインテグレーションの最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. メッセージの公開に使用する AMQP コネクションをクリックします。
3. **Choose an action** ページで、以下のアクションの1つを選択します。
 - **Publish messages** で応答を受信せずに、指定するキューまたはトピックへメッセージを公開します。このアクションを設定するには、以下を行います。
 - a. **Destination name** フィールドに、メッセージの送信先となるキューまたはトピックの名前を入力します。
 - b. **Destination type** では **Queue** を許可するか、**Topic** を選択します。

- c. **Persistent** を選択し、コネクションが失敗した場合でもメッセージが確実に送信されるようにします。
- **Request response using messages** で、指定するキューまたはトピックへメッセージをパブリッシュし、応答を受信します。



注記

シンプルなインテグレーションの最後のコネクションが **Request response using messages** アクションを実行する AMQP コネクションである場合、コネクションはメッセージをパブリッシュしますが、応答は破棄されます。応答を保持するには、**Request response using messages** アクションを実行する途中のコネクションとして AMQP コネクションを追加し、ログステップでシンプルなインテグレーションを終了します。

このアクションを設定するには、以下を行います。

- a. **Destination name** フィールドに、メッセージの送信先となるキューまたはトピックの名前を入力します。
- b. **Destination type** では **Queue** を許可するか、**Topic** を選択します。
- c. メッセージを見逃さずにコネクションを開閉できるようにするため、**Durable subscription ID** フィールドに永続サブスクリプション ID を入力します。宛先のタイプはトピックにする必要があります。
- d. 特定の条件を満たす応答のみを受信する場合は、**Message selector** フィールドにフィルター式を入力します。
メッセージセレクトターは式が含まれる文字列です。式の構文は、SQL92 条件式構文のサブセットを基にしています。以下の例のメッセージセレクトターは、値が **Sports** または **Opinion** に設定された **NewsType** プロパティがあるメッセージをすべて選択します。

NewsType = 'Sports' OR NewsType = 'Opinion'

メッセージコンシューマーは、ヘッダーとプロパティがメッセージセレクトターの式と一致するメッセージのみを受信します。メッセージセレクトターは、メッセージのボディ部分の内容を基にしてメッセージを選択することはできません。

- 4. **Next** をクリックし、アクションの入力および出力タイプを指定します。
- 5. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。

- **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
6. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 7. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 8. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 9. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

第5章 BOX への接続

インテグレーションでは、Box からファイルをダウンロードしたり、Box にファイルをアップロードしたりすることができます。

Box への接続はシンプルなインテグレーションを開始できません。つまり、Box 接続を使用してインテグレーションの実行をトリガーすることはできません。



重要

Box への接続はテクノロジープレビューの機能です。テクノロジープレビュー機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、Red Hat は本番環境での使用は推奨しません。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能により、近日発表予定の製品機能をリリースに先駆けてご提供でき、お客様は開発プロセス時に機能をテストして、フィードバックをお寄せいただくことができます。Red Hat のテクノロジープレビュー機能のサポート範囲に関する詳細は、<https://access.redhat.com/ja/support/offerings/techpreview> を参照してください。

詳細は以下を参照してください。

- 「Fuse Online を Box クライアントとして登録」
- 「Box 接続の作成」
- 「インテグレーションの途中での Box ファイルのダウンロードまたはアップロード」
- 「Box にファイルをアップロードしてインテグレーションを終了」

5.1. FUSE ONLINE を BOX クライアントとして登録


Fuse Online 環境を Box にアクセスできるクライアントアプリケーションとして登録する必要があります。これにより、Box に接続するインテグレーションをいくつでも作成することができます。そのため、特定の Fuse Online 環境を 1 度だけ Box に登録する必要があります。

Fuse Online を Box クライアントアプリケーションとして登録できるのは、Fuse Online 環境ごとに 1 回のみです。しかし、各 Box 接続は同じ登録を使用しますが、異なるユーザークレデンシャルを使用することができます。

前提条件

- <https://app.box.com/developers/console> で取得できる Box 開発者アカウントを持っている必要があります。

手順

1. Fuse Online で以下を行います。
 - a. 左側のパネルで **Settings** をクリックします。
 - b. **Settings** ページで、上部付近にあるコールバック URL の右側で  をクリックし、Fuse Online 環境のコールバック URL をクリップボードにコピーします。この手順の最後の方でこの URL が必要になります。

2. 別のブラウザタブで <https://app.box.com/developers/console> の Box 開発者ポータルにアクセスし、以下を行います。
 - a. ログインします。
 - b. **Create New App** をクリックします。
 - c. **Custom App** をクリックし、**Next** をクリックします。
 - d. **Authentication Method** で **Standard OAuth 2.0 (User Authentication)** をクリックし、**Next** をクリックします。
 - e. アプリケーション名 (例: **Fuse Online Client**) を入力し、**Create App** をクリックします。
 - f. **View Your App** をクリックします。
 - g. 任意手順:**OAuth 2.0 Credentials** セクションで、クライアント ID とクライアントシークレットを安全な場所にコピーします。Box へのコネクションを作成するには、これらの値が必要です。
 - h. **Save Changes** をクリックします。

結果

これで Fuse Online 環境が Box クライアントとして登録され、Fuse Online はログインした Box アカウ
ントのコンテンツにアクセスできるようになります。

5.2. BOX コネクションの作成

インテグレーションで Box ファイルをダウンロードまたはアップロードするには、インテグレーション
に追加できる Box コネクションを作成します。同じコネクションを任意の数のインテグレーションに追
加できます。

前提条件

- Fuse Online 環境を Box にアクセスできるアプリケーションとして登録する Box クライアント
アプリケーションが必要です。
- そのアプリケーションを作成した Box 開発者アカウントにログインできるか、そのアプリケー
ションの Box クライアント ID および Box クライアントシークレットを取得している必要があ
ります。
- Box にアクセスするためにこのコネクションが使用する Box アカウントのユーザー名およびパ
スワードを知っている必要があります。

手順

1. すでに Box クライアント ID とクライアントシークレットがある場合は、次のステップを省略
します。ない場合は、新しいブラウザタブで <https://app.box.com/developers/console> にア
クセスし、以下を行います。
 - a. Fuse Online 環境から Box へのアクセスを登録するアプリケーションを作成した Box アカ
ウトにログインします。
 - b. **My Apps** ページで、Fuse Online アプリケーションをクリックして設定を表示します。
 - c. 左側の **Configuration** をクリックします。

- d. **OAuth 2.0 Credentials** セクションで、クライアント ID をクリップボードにコピーします。
2. 別のブラウザタブを使用して、Fuse Online で以下を行います。
 - a. **Connections** をクリックします。
 - b. **Create Connection** をクリックします。
 - c. **Box** コネクタをクリックします。
 - d. **User name** フィールドに、Box ファイルをダウンロードまたはアップロードするためにこの接続が使用する Box アカウントのユーザー名を入力します。
 - e. **User password** フィールドに、そのアカウントのパスワードを入力します。
 - f. **Client ID** フィールドに Box のクライアント ID を貼り付けます。
 3. 任意手順:必要な場合は Box アプリケーションの **OAuth 2.0 Credentials** に戻り、クライアントシークレットをクリップボードにコピーして Fuse Online に戻ります。
 4. **Client secret** フィールドに Box のクライアントシークレットの文字列を貼り付けます。
 5. **Validate** をクリックします。Fuse Online はこの接続を検証できるかを示すメッセージを表示します。検証に失敗した場合は、入力した値が正しいことを確認し、再度検証を行ってください。
 6. 検証に成功したら **Next** をクリックします。
 7. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**Box Sales Account** を入力します。
 8. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。たとえば、**Sample Box connection that can access content in our company sales Box account.** を入力します。
 9. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **Box Sales Account** が表示されます。

5.3. インテグレーションの途中での **BOX** ファイルのダウンロードまたはアップロード

インテグレーションの途中で Box からファイルをダウンロードしたり Box にファイルをアップロードしたりするには、Box コネクションをフローの途中に追加します。

前提条件

- Box コネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。

2. インテグレーションが使用する Box コネクションをクリックします。
3. **Choose an action** ページで、以下の1つを選択します。
 - このコネクションがアクセスする Box アカウントからファイルを取得する場合は **Download** を選択します。
 - このコネクションがアクセスする Box アカウントにファイルをアップロードする場合は **Upload** を選択します。
4. 選択したアクションを設定します。

ファイルを **ダウンロード** する場合は以下を行います。

 - a. **File encoding** フィールドで **UTF-8** を許可するか、コネクションが取得するファイルのエンコーディングを指定します。
 - b. 任意手順:**File ID** フィールドで、取得するファイルの Box ID を指定します。ファイル ID は、Box のファイルを表示する場合に URL の最後にあります。たとえば、URL が **https://app.box.com/file/537364588548** の場合、**537364588548** がファイル ID になります。

この代わりに、このコネクションの前にデータマッピングステップを追加し、前のステップのファイル ID をこのコネクションにマップできます。

本リリースでは、単一のファイルのみをダウンロードできます。しかし、前のステップからファイル ID をマップすると、ファイルの識別を動的にすることができます。たとえば、ID はデータベースから取得することができます。そのため、実行ごとに異なるファイルをダウンロードする可能性があります。

ファイルを **アップロード** する場合は以下を行います。

- a. **Parent folder ID** フィールドに、ファイルのアップロード先となる Box フォルダーの ID を入力します。

Box のフォルダーの内容を表示している場合、URL の最後が Box フォルダー ID になります。たとえば URL が **https://app.box.com/folder/89490291417** の場合、フォルダー ID は **89490291417** になります。本リリースでは、単一のファイルのみをアップロードできます。
- b. 任意手順:**File name** フィールドに新規ファイルの名前を入力します。指定した Box フォルダーにすでに存在するファイルの名前を指定すると、ランタイムエラーが発生します。

コネクションへの入力、Box にアップロードするファイルである必要があります。ファイル名を指定しない場合、コネクションへの入力ファイルと同じ名前のファイルがすでに存在しなければ、それと同じ名前のファイルをアップロードします。
- c. **Next** をクリックして、アップロードアクションの入力タイプを指定します。
- d. **Select Type** フィールドで **Type specification not required** を指定し、**Next** をクリックします。

結果

インテグレーションフローで追加した場所にコネクションが表示されます。

5.4. BOX にファイルをアップロードしてインテグレーションを終了

ファイルを Box にアップロードしてインテグレーションを終了するには、Box コネクションをインテグレーションの最後のコネクションとして追加します。

前提条件

- Box コネクションが作成されている必要があります。
- シンプルなインテグレーションを作成または編集することになります。Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Choose a connection** ページで、インテグレーションの終了に使用する Box コネクションをクリックします。
2. **Choose an action** ページで **Upload** アクションを選択し、このコネクションがアクセスする Box アカウントにファイルをアップロードします。
3. **Parent folder ID** フィールドに、ファイルのアップロード先となる Box フォルダーの ID を入力します。
Box のフォルダーを選択する場合、URL の最後が Box のフォルダー ID になります。たとえば URL が **https://app.box.com/folder/89490291417** の場合、フォルダー ID は **89490291417** になります。本リリースでは、単一のファイルのみをアップロードできます。
4. 任意手順:**File name** フィールドに新規ファイルまたは既存ファイルの名前を入力します。指定した Box フォルダーにすでに存在するファイルの名前を指定すると、コネクションによってコンテンツがそのファイルに追加されます。
コネクションへの入力は、Box にアップロードするファイルである必要があります。ファイル名を指定しない場合、コネクションへの入力ファイルと同じ名前のファイルをアップロードします。
5. **Next** をクリックします。
6. **Select Type** フィールドで **Type specification not required** を指定し、**Next** をクリックします。

結果

インテグレーションビジュアライゼーションの最後にコネクションが表示されます。

第6章 DROPBOX への接続

インテグレーションでは、Dropbox からファイルをダウンロードしたり、Dropbox にファイルをアップロードすることができます。詳細は以下を参照してください。

- 「Fuse Online を Dropbox クライアントとして登録」
- 「Dropbox コネクションの作成」
- 「Dropbox からファイルを取得してインテグレーションの実行をトリガー」
- 「Dropbox にファイルを追加してインテグレーションを終了」
- 「インテグレーションの途中で Dropbox にアクセス」

6.1. FUSE ONLINE を DROPBOX クライアントとして登録

Fuse Online 環境を Dropbox にアクセスできるクライアントアプリケーションとして登録する必要があります。これにより、Dropbox に接続するインテグレーションをいくつでも作成することができます。そのため、特定の Fuse Online 環境を1度だけ Dropbox に登録する必要があります。

Fuse Online を Dropbox クライアントアプリケーションとして登録できるのは、Fuse Online 環境ごとに1回のみです。しかし、各 Dropbox コネクションは同じ登録を使用しますが、異なるユーザークレデンシャルを使用することができます。

前提条件

ファイルのダウンロードまたはアップロードでインテグレーションが使用する Dropbox アカウントにログインできる必要があります。

手順

1. Fuse Online で以下を行います。
 - a. 左側のパネルで **Settings** をクリックします。
 - b. **Settings** ページで、上部付近にあるコールバック URL の右側で  をクリックし、Fuse Online 環境のコールバック URL をクリップボードにコピーします。この手順の最後の方でこの URL が必要になります。
2. 別のブラウザタブで <https://www.dropbox.com> にアクセスし、以下を行います。
 - a. インテグレーションでアクセスするデータがある Dropbox アカウントにログインします。
 - b. ログインした後に <https://www.dropbox.com/developers/apps> にアクセスします。
 - c. **Create App** をクリックします。
 - d. **Dropbox API** を選択します。
 - e. この手順の最初にクリップボードにコピーした URL を、ページの上付近にある **During registration, enter this callback URL:** で始まる文に貼り付けます。たとえば、貼り付ける URL は `https://app-proj9128.7b63.fuse-ignite.openshiftapps.com/api/v1/credentials/callback` のようになります。

- f. Fuse Online が単一のフォルダーにアクセスするか、またはすべてのフォルダーやファイルにアクセスするかを選択します。
- g. Dropbox アプリケーションの名前を指定します。たとえば、**Fuse Online Access From Aslan LLC** のような名前を指定できます。Dropbox アプリケーションで一意的な名前を指定する必要があります。
- h. ボックスにチェックマークを入れ、Dropbox API の利用規約に同意します。
- i. **Create App** をクリックします。
- j. 新しいアプリケーションの Dropbox **Settings** ページにある **OAuth2 Redirect URIs** の入力フィールドに、この手順の始めにクリップボードにコピーした Fuse Online URL を貼り付けます。
- k. **Add** をクリックします。

結果

これで Fuse Online 環境が Dropbox クライアントとして登録され、Fuse Online はログインした Dropbox アカウントのコンテンツにアクセスできるようになります。

6.2. DROPBOX コネクションの作成

インテグレーションで Dropbox ファイルをダウンロードまたはアップロードするには、インテグレーションに追加できる Dropbox コネクションを作成します。同じコネクションを任意の数のインテグレーションに追加できます。

前提条件

Fuse Online 環境が Dropbox にアクセスできるアプリケーションとして登録されている必要があります。

手順

1. 新しいブラウザのタブで <https://www.dropbox.com> にアクセスし、以下を行います。
 - a. Fuse Online 環境からのアクセスを登録するアプリケーションを作成した Dropbox アカウントにログインします。
 - b. <https://www.dropbox.com/developers/apps> にアクセスします。
 - c. Fuse Online アプリケーションをクリックし、その設定を表示します。
2. Fuse Online の別のブラウザタブで以下を行います。
 - a. 左側のパネルで **Connections** をクリックし、利用できるコネクションを表示します。
 - b. 右上の **Create Connection** をクリックし、利用可能なコネクターを表示します。
 - c. **Dropbox** コネクターをクリックします。
3. アプリケーションの Dropbox 設定表示に戻り、以下を行います。
 - a. 下にスクロールして **Generated Access Token** を確認します。
 - b. **Generate** をクリックします。

- c. 生成されたアクセストークンをクリップボードにコピーします。
4. Fuse Online に戻り、**Configure Connection** ページの **Access Token** フィールドに生成されたアクセストークンを貼り付けます。
5. **Client Identifier** フィールドに Dropbox アプリケーションを作成したときに指定した名前を入力します。
6. **Validate** をクリックします。Fuse Online はこのコネクションを検証できるかを示すメッセージを表示します。検証に失敗した場合は、入力した値が正しいことを確認し、再度検証を行ってください。
7. 検証に成功したら **Next** をクリックします。
8. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Dropbox Connect 1** を入力します。
9. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample Dropbox connection that can access all content in our company Dropbox account.** を入力します。
10. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Dropbox Connect 1** が表示されます。

6.3. DROPBOX からファイルを取得してインテグレーションの実行をトリガー

Dropbox からファイルをダウンロードしてインテグレーションを開始するには、Dropbox コネクションを最初のコネクションとして追加します。

前提条件

Dropbox コネクションが作成されている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose an action** ページで、インテグレーションを開始するために使用する Dropbox コネクションをクリックします。
4. **Choose an action** ページで **Download** アクションを選択し、このコネクションがアクセスする Dropbox アカウントから1つまたは複数のファイルを取得します。
5. アクションを設定するには、**Folder or file name path to download** フィールドにインテグレーションが取得するコンテンツのファイル名のパスを指定します。本リリースでは、単一のファイルのみをダウンロードできます。
6. **Next** をクリックして、アクションの出力タイプを指定します。
7. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。

これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。

- **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
8. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 9. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 10. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 11. **Next** をクリックします。

結果

コネクションによってシンプルなインテグレーションが開始され、Fuse Online で最後のコネクションを選択するよう要求されます。

6.4. DROPBOX にファイルを追加してインテグレーションを終了

ファイルを Dropbox にアップロードしてインテグレーションを終了するには、Dropbox コネクションをインテグレーションの最後のコネクションとして追加します。

前提条件

- Dropbox コネクションが作成されている必要があります。
- インテグレーションを作成または編集することになります。Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Choose a connection** ページで、インテグレーションの終了に使用する Dropbox コネクションをクリックします。
2. **Choose an action** ページで **Upload** アクションを選択し、このコネクションがアクセスする Dropbox アカウントに現在のインテグレーションデータを追加します。

3. **Remote Path** フィールドにアップロードするファイルのローカルファイル名パスを入力します。Dropbox は同じパスと名前ファイルでファイルを格納します。本リリースでは、単一のファイルのみをアップロードできます。
4. **Upload mode** では以下を行います。
 - 同じ Dropbox フォルダーに同じ名前のファイルがない場合のみ **Add** を選択してファイルをアップロードします。同じ名前のファイルが同じ Dropbox フォルダーに存在する場合は、ファイルはアップロードされず、インテグレーションが継続されます。これは、アップロードするファイルのコンテンツが更新されたかどうかに関わらず、同じ動作になります。
 - 同じ名前のファイルが同じ Dropbox フォルダーにある場合でも確実にファイルをアップロードするには **Force** を選択します。既存のファイルはアップロードするファイルによって上書きされます。
5. **Next** をクリックしてアクションの入力タイプを指定します。
6. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
7. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
8. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
9. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
10. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションの最後にコネクションが表示されます。

6.5. インテグレーションの途中で DROPBOX にアクセス

フローの途中で Dropbox にファイルをアップロードするには、Dropbox コネクションをフローの途中に追加します。

前提条件

- Dropbox コネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. インテグレーションが使用する Dropbox コネクションをクリックします。
3. **Choose an action** ページで **Upload** を選択し、このコネクションがアクセスする Dropbox アカウントに現在のインテグレーションデータを追加します。
4. このアクションを設定するには以下を行います。
 - a. **Remote Path** フィールドにアップロードするファイルのローカルパスとファイル名を指定します。Dropbox は同じパスと名前ファイルでファイルを格納します。本リリースでは、単一のファイルのみをアップロードできます。
 - b. **Upload mode** では以下を行います。
 - 同じ Dropbox フォルダーに同じ名前のファイルがない場合のみ **Add** を選択してファイルをアップロードします。同じ名前のファイルが同じ Dropbox フォルダーに存在する場合は、ファイルはアップロードされず、インテグレーションが続行されます。これは、アップロードするファイルのコンテンツが更新されたかどうかに関わらず、同じ動作になります。
 - 同じ名前のファイルが同じ Dropbox フォルダーにある場合でも確実にファイルをアップロードするには **Force** を選択します。既存のファイルはアップロードするファイルによって上書きされます。
5. **Next** をクリックし、アクションの入力および出力タイプを指定します。
6. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。

7. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
8. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアルライゼーションとデータマッパーにタイプ名が表示されます。
9. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
10. **Next** をクリックします。

結果

インテグレーションフローで追加した場所にコネクションが表示されます。

第7章 メールサーバーへの接続 (IMAP、POP3、SMTP)

インテグレーションは、IMAP または POP3 メールサーバーに接続してメールを取得したり、SMTP メールサーバーに接続してメールを送信したりすることができます。メールの取得時にインテグレーションの実行をトリガーするには、IMAP または POP3 メールサーバーへのコネクションを作成し、そのコネクションをシンプルなインテグレーションの最初のコネクションとして追加します。フローからメールを送信するには、SMTP メールサーバーへのコネクションを作成し、そのコネクションをフローの途中に追加するか、シンプルなインテグレーションを終了するために追加します。

詳細は、以下を参照してください。

- [「IMAP または POP3 メールサーバーへのコネクションの作成」](#)
- [「SMTP メールサーバーへのコネクションの作成」](#)
- [「メールを取得してインテグレーションの実行をトリガー」](#)
- [「インテグレーションからのメールの送信」](#)

7.1. IMAP または POP3 メールサーバーへのコネクションの作成

受信したメールによって実行がトリガーされるシンプルなインテグレーションを作成する前に、IMAP または POP3 メールサーバーへのコネクションを作成する必要があります。

前提条件

このコネクションによってアクセスされるメールサーバーについて熟知している必要があります。特に、以下を知っている必要があります。

- メールサーバーが IMAP または POP3 プロトコルを使用するかどうか。
- 認証クレデンシャルが必要かどうか。必要な場合、ユーザー名とパスワードが必要です。
- セキュリティーメソッドが実装されているかどうか。実装されている場合はどのメソッドかを知っている必要があります。
- サーバー証明書が必要かどうか。必要な場合、サーバー証明書を指定できる必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクターを表示します。
3. **Receive Emails** コネクターをクリックします。
4. コネクションを設定します。
 - a. **Protocol** フィールドで、**imap** を使用するか、**pop3** を選択します。
 - b. **Email Host Name** フィールドに、メールサーバーのホスト名、仮想ホスト名、または IP アドレスを入力します。
 - c. **Email Server Port Number** フィールドに、メールサーバーがリスンするポート番号を入力します。

- d. **User Name** フィールドに、この接続のアクセスが許可されるメールアカウントのユーザー名を指定します。この接続は、インテグレーションの実行中にこのメールアカウントをポーリングしてメッセージを受信します。
まれに、メールサーバーがアクセスを認証しないことがあります。このような場合、ユーザー名やパスワードを指定する必要はありません。
 - e. **Password** フィールドに、この接続のアクセスが許可されるメールアカウントのパスワードを指定します。
 - f. 任意手順:メールサーバーがセキュリティーメソッドを実装する場合は **Security Method** フィールドで **StartTLS** または **SSL/TLS** を選択します。
 - g. 任意手順:メールサーバーが内部ネットワークにある非公開のサーバーで、セキュリティーとして **StartTLS** または **SSL/TLS** を選択した場合、**Server Certificate** フィールドに自己署名メールサーバー証明書を貼り付けます。サーバー証明書が必要な場合、証明書を指定しないと Fuse Online はこの接続を検証できません。
5. **Validate** をクリックします。Fuse Online は即座に接続の検証を試み、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
 6. 検証に成功したら **Next** をクリックします。
 7. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**Obtain Info Emails** を入力します。
 8. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。たとえば、**Connect to our company email server to obtain messages sent to info@ourcompany.com.** を入力します。
 9. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **Obtain Info Emails** が表示されます。

7.2. SMTP メールサーバーへの接続の作成

SMTP メールサーバーに接続してメールを送信するため、インテグレーションフローを編集する前に SMTP メールサーバーへの接続を作成する必要があります。

前提条件

この接続によってアクセスされるメールサーバーについて熟知している必要があります。特に、以下を知っている必要があります。

- 認証クレデンシャルが必要かどうか。必要な場合、ユーザー名とパスワードが必要です。
- セキュリティーメソッドが実装されているかどうか。実装されている場合はどのメソッドかを知っている必要があります。
- サーバー証明書が必要かどうか。必要な場合、サーバー証明書を指定できる必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能な接続を表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクターを表示します。

3. **Send Email** コネクタをクリックします。
4. コネクションを設定します。
 - a. **Email Host Name** フィールドに、メールサーバーのホスト名、仮想ホスト名、または IP アドレスを入力します。
 - b. **Email Server Port Number** フィールドに、メールサーバーがリスンするポート番号を入力します。
 - c. **User Name** フィールドに、このコネクションのアクセスが許可されるメールアカウントのユーザー名を指定します。このコネクションは、インテグレーションの実行中にこのメールアカウントを使用してメッセージを送信します。
まれに、メールサーバーがアクセスを認証しないことがあります。このような場合、ユーザー名やパスワードを指定する必要はありません。
 - d. **Password** フィールドに、このコネクションのアクセスが許可されるメールアカウントのパスワードを指定します。
 - e. 任意手順:メールサーバーがセキュリティーメソッドを実装する場合は **Security Method** フィールドで **StartTLS** または **SSL/TLS** を選択します。
 - f. 任意手順:メールサーバーが内部ネットワークにある非公開のサーバーで、セキュリティーとして **StartTLS** または **SSL/TLS** を選択した場合、**Server Certificate** フィールドに自己署名メールサーバー証明書を貼り付けます。サーバー証明書が必要な場合、証明書を指定しないと Fuse Online はこのコネクションを検証できません。
5. **Validate** をクリックします。Fuse Online は即座にコネクションの検証を試み、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
6. 検証に成功したら **Next** をクリックします。
7. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Send Emails to Leads** を入力します。
8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Connect to our company email server to send messages from info@ourcompany.com.** を入力します。
9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Send Emails to Leads** が表示されます。

7.3. メールを取得してインテグレーションの実行をトリガー

メールの受信時にインテグレーションの実行をトリガーするには、IMAP または POP3 メールサーバーコネクションをシンプルなインテグレーションの最初のコネクションとして追加します。

前提条件

- メールを取得元であるアカウントにアクセスできるように設定された IMAP または POP3 メールサーバーコネクションが作成されている必要があります。
- メッセージを取得するフォルダーの名前を知っている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用するメールサーバーコネクションをクリックします。
4. **Choose an action** ページで、**Receive Email** を選択し、コネクションのアクセスが許可されるメールアカウントからメッセージを定期的に取得します。
5. アクションを設定するには以下を行います。
 - a. **Folder** フィールドを空白のままにして受信トレイからメールを取得します。特定のフォルダーからメッセージを取得する場合は、そのフォルダーの名前を入力します。コネクションがPOP3 メールサーバーにアクセスする場合、**Inbox** (受信トレイ) や **Sent** (送信済み) などの標準のフォルダーのみがサポートされます。
 - b. **Unseen Only** チェックボックスを選択すると、メールサーバーで未読となっているメッセージのみを取得します。
 - c. **Delay** フィールドで、デフォルトの5秒をメッセージのポーリング間隔の経過時間として指定します。別のポーリング間隔を指定する場合は、数値を入力し、時間の単位を選択します。
 - d. **Maximum Emails** フィールドで、1回のポーリング操作が返すことが可能な最大メッセージ数を入力します。デフォルトは5です。受信トレイまたはフォルダーに返信可能な最大数を超えるメッセージが含まれる場合、コネクションは最も古いメッセージを **Maximum Emails** に設定された数まで返します。ポーリングごとに以下を取得する場合は、**Maximum Emails** を **-1** に設定します。
 - **Unseen Only** が選択されている場合、指定フォルダーの新しいメッセージすべてを取得。
 - **Unseen Only** が選択されていない場合、指定フォルダーのメッセージすべてを取得。
6. **Next** をクリックします。

結果

インテグレーションフローの最初にコネクションが表示されます。実行中にポーリングがメールを返した場合、コネクションはメッセージをインテグレーションの次のステップに渡します。ポーリングが返すメールがない場合、インテグレーションは実行を継続しますが、Fuse Online は残りのインテグレーションステップをトリガーしません。

例

以下のように設定された **Receive Email** アクションを例として取り上げます。

- **Folder** フィールドを空白にして、コネクションが受信トレイをポーリングするようにします。
- **Unseen Only** を選択します。
- デフォルトの **Delay** (5秒) および **Maximum Emails** (5) を指定します。

遅延中、次のポーリングが行われる前に10個のメッセージが受信トレイに受信されたとします。コネクションは最も古い5つのメッセージを返します。5秒後に次のポーリングが行われ、コネクションは残りの5つのメッセージを返します。**Maximum Emails** に設定された数よりも多くのメッセージがある

場合、コネクションは古いメッセージから返すため、新しいメッセージが受信されても残りの5つのメッセージが返されます。

7.4. インテグレーションからのメールの送信

メッセージを送信するメールサーバーコネクションは、フローの途中に追加したり、シンプルなインテグレーションを終了するために追加することができます。

前提条件

- メッセージの送信元となるアカウントにアクセスできるように設定された SMTP メールサーバーコネクションが作成されている必要があります。
- Fuse Online はインテグレーションの追加を要求したり、シンプルなインテグレーションの最後のコネクションを選択するよう要求します。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. メッセージの送信に使用するメールサーバーコネクションをクリックします。
3. **Send Email** アクションを選択します。
4. メッセージを送信するメールアドレスを **Email from** フィールドに入力します。これは、メールアドレスを前のインテグレーションステップからこのコネクションにマップするデータマップステップを追加する場合でも必要です。
5. 推奨手順: 以下の各フィールドに値を指定します。複数のメールアドレスはコンマで区切ります。
 - a. **Email to**
 - b. **Email subject**
 - c. **Email text**
 - d. **Email cc**
 - e. **Email bcc**
6. **Parameter Priority** フィールドで **Input Values** を指定するか、**Consumed Data** を選択します。これにより、以下のいずれかが優先されることを決定します。
 - このアクション設定の入力値
 - このコネクションにマップするデータ値

Consumed Data の選択が推奨されます。これにより、メールフィールドに指定する値がデフォルトの値になります。
7. **Next** をクリックして、コネクションをフローに追加します。

結果

インテグレーションフローで追加した場所に接続が表示されます。

次のステップ

必要な他の接続をすべてフローに追加します。その後、メールを送信する接続の前にデータマッピングステップを追加します。データマッピングステップでは、メールを送信するためにソースフィールドをターゲットフィールドにマップします。

第8章 FHIR への接続

FHIR (Fast Healthcare Interoperability Resources) は医療データを交換するための標準です。インテグレーションでは、1つ以上の FHIR リソースの取得、リソースの作成または更新、リソースの1つ以上のフィールドの更新、またはトランザクションを使用した複数リソースの作成を行うことができます。これには、フローの途中で FHIR コネクションを追加するか、FHIR コネクションをシンプルなインテグレーションの最後のコネクションとして追加します。

FHIR への Fuse Online コネクションは以下を行うことができます。

- Basic または Bearer トークン (OAuth) 認証を使用する FHIR サーバーと通信できます。
- FHIR DSTU3 に準拠するすべての FHIR リソースで操作することができます。
- リソースにカスタムデータを格納できるようにするエクステンションでは操作できません。FHIR サーバーから受信したデータにエクステンションがある場合、インテグレーションは何もせずに無視します。

FHIR に接続するインテグレーションでは、データマッパーステップで FHIR リストフィールドに対して1レベルの深さのみをマッピングできます。そのため、FHIR リストにリストが含まれる場合は、入れ子のリストでフィールドをマッピングできません。この制限を軽減するため、Fuse Online は一部の FHIR リソースリストフィールドを単一値のフィールドに変換します。たとえば、リストフィールドのアドレスはデータマッパーで単一のフィールドとして表示される可能性があります。

FHIR への接続に関する詳細は、以下を参照してください。

- [「FHIR サーバーへのコネクションの作成」](#)
- [「FHIR サーバーからのリソースの取得」](#)
- [「リソースの FHIR サーバーのクエリー」](#)
- [「FHIR サーバーでのリソースの作成」](#)
- [「FHIR サーバー上のリソースにあるすべてのフィールドを更新」](#)
- [「FHIR サーバー上のリソースにある指定のフィールドを更新」](#)
- [「FHIR サーバーで異なるタイプのリソースの作成」](#)
- [「FHIR サーバーからのリソースの削除」](#)

8.1. FHIR サーバーへのコネクションの作成

インテグレーションで FHIR リソースを操作するには、FHIR サーバーへのコネクションを作成し、そのコネクションをフローの途中で追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

FHIR コネクションはシンプルなインテグレーションの最初のコネクションとすることはできません。しかし、FHIR コネクションを定期的にトリガーするタイマーを使ってシンプルなインテグレーションを開始することができます。

前提条件

- 接続する FHIR サーバーの URL を知っている必要があります。

- FHIR サーバーへアクセスするための承認クレデンシャルを持っている必要があります。これらのクレデンシャルは FHIR サーバー管理者から取得する必要があります。まれに、FHIR サーバーが認証を必要とせず、クレデンシャルを指定せずに接続を作成できることがあります。たとえば、パブリック FHIR サーバーやプライベートネットワークの FHIR サーバーが認証を必要としないことがあります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能な接続を表示します。
2. 右上の **Create Connection** をクリックしてコネクタを表示します。
3. **FHIR コネクタ** をクリックし、以下のように接続を設定します。
 - a. **FHIR version** フィールドで必要となる **DSTU3** を許可します。
 - b. アクセスする FHIR サーバーの URL を **FHIR server URL** フィールドに入力します。インテグレーションの作成中、デフォルトテストサーバー **http://fhirtest.uhn.ca/baseDstu3** を許可することができます。
 - c. **FHIR server username** フィールドにユーザー名を入力します。これは、Basic 認証を使用する FHIR サーバーで必要になります。
 - d. **FHIR server password** フィールドにパスワードを入力します。これも Basic 認証を使用する FHIR サーバーで必要になります。
 - e. **FHIR server bearer token** フィールドにトークンを入力します。これは、OAuth 認証を使用する FHIR サーバーで必要になります。
4. **Validate** をクリックします。Fuse Online は即座に接続を検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、入力パラメーターを訂正し、再度検証を行います。
5. 検証に成功した場合は **Next** をクリックします。
6. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**FHIR West** を入力します。
7. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。
8. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **FHIR West** が表示されます。

8.2. FHIR サーバーからのリソースの取得

インテグレーションでは、指定タイプのリソースを1つ取得できます。これには、フローの途中で FHIR コネクションを追加します。

前提条件

- 取得するリソースを持つ FHIR サーバーへの接続が作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。
2. 取得するリソースにアクセスできる FHIR コネクションをクリックします。
3. **Choose an action** ページで **Read** を選択します。
4. **Resource Type** フィールドをクリックし、FHIR リソースタイプのリストを表示します。取得するリソースのタイプを選択または入力します。
5. 取得するリソースタイプに他のリソースタイプが含まれている場合、**Contained Resource Types** フィールドで含まれているタイプを選択します。複数のタイプを選択する必要がある場合は **Ctrl** キーを使用します。
含まれるリソースタイプを特定すると、データマッパーは含まれるリソースにあるフィールドを表示できるようになります。含まれるリソースタイプを指定しないと、含まれるリソースにあるフィールドからマッピングを行うことができません。実際に含まれないリソースを選択しても問題はありません。
6. **Resource Id** フィールドにリソースの ID を入力します。ただし、フローの初期のステップからマップする予定である場合は、このフィールドを空白のままにしておきます。
7. **Resource version** フィールドには、リソースのバージョン ID を任意で指定します。最新バージョンのリソースを取得する場合や、以前のステップからリソースのバージョン ID をマップする場合は、このフィールドを空白のままにしておきます。
8. **Next** をクリックして、この接続をフローに追加します。

結果

インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。実行中、この接続は1つのリソースを返します。

次のステップ

Resource Id フィールドまたは **Resource version** フィールドに値をマップするには、この接続の前にデータマッパーステップを追加します。

8.3. リソースの FHIR サーバーのクエリー

インテグレーションでは、FHIR サーバーにクエリーを行って、指定のクエリーを満たす特定の FHIR リソースのインスタンスを取得できます。たとえば、65 歳以上の患者全員が肺炎の予防接種を受けるようにするには、1955 年より前に誕生した患者のリソースを取得するクエリーを指定します。1955 年より前に禁止されていた患者のリソースを取得するクエリーを指定します。これには、フローの途中に FHIR コネクションを追加します。

前提条件

- 取得するリソースを持つ FHIR サーバーへの接続が作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。

2. 取得するリソースを持つ FHIR コネクションをクリックします。
3. **Choose an action** ページで **Search** を選択します。
4. **Resource Type** フィールドをクリックし、FHIR リソースタイプのリストを表示します。取得するリソースのタイプを選択または入力します。
5. **Query** フィールドに FHIR クエリーを指定するか、このフィールドを空白のままにして以前のステップからクエリーをマッピングします。FHIR クエリーの作成に関する詳細は、[FHIR Release 3 Search](#) を参照してください。
6. **Next** をクリックして、このコネクションをフローに追加します。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。実行中、このコネクションは指定のクエリーを満たすリソースのリストを返します。Fuse Online はこのリストをコレクションとして扱います。

次のステップ

クエリーを指定しなかった場合は、このコネクションの前にデータマッパーステップを追加し、クエリーをこのコネクションにマップします。返されたリストのコレクションではなく、返された個々のリソースで操作を行うには、フローのこのコネクションの後に分割ステップ (split step) を追加します。

8.4. FHIR サーバーでのリソースの作成

インテグレーションではリソースを作成し、FHIR サーバーに追加することができます。コネクションの **Create** アクションを設定するときに、作成するリソースのタイプを指定します。たとえば、新しい患者のリソースを FHIR サーバーに追加する場合があります。リソースを作成するには、フローの途中で FHIR コネクションを追加するか、FHIR コネクションをシンプルなインテグレーションの最後のコネクションとして追加します。

新しいリソースに値を入力するには、このステップを追加する FHIR コネクションのすぐ前にデータマッパーステップを追加します。フローの以前のステップは、新しいリソースに含まれるデータを提供することが想定されます。この FHIR コネクションが作成するターゲットリソースのフィールドに、以前のステップのフィールドをマップします。新しいリソースにはマップしたフィールドのみが含まれます。

コネクションが **Create** アクションを実行して FHIR サーバー上で新しいリソースを作成する場合、新しいリソースには自動生成された FHIR リソース ID が割り当てられます。新しいリソースのリソース ID を指定する場合は、**Create** アクションを選択する代わりに、コネクションが実行する **Update アクション** を選択します。

前提条件

- リソースを追加する FHIR サーバーへのコネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、最後のコネクションを追加するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。

2. リソースを追加するサーバーへの FHIR コネクションをクリックします。
3. **Choose an action** ページで **Create** を選択します。
4. **Resource Type** フィールドをクリックし、FHIR リソースタイプのリストを表示します。作成するリソースのタイプを選択または入力します。
5. 作成するリソースタイプに他のリソースタイプが含まれている場合、**Contained Resource Types** フィールドで含まれているタイプを選択します。複数のタイプを選択する必要がある場合は **Ctrl** キーを使用します。
含まれるリソースタイプを特定すると、データマッパーは含まれるリソースにあるフィールドを表示できるようになります。含まれるリソースタイプを指定しないと、含まれるリソースにあるフィールドをマップ元またはマップ先としてマッピングすることができません。実際に含まれないリソースを選択しても問題はありません。
6. **Next** をクリックして、このコネクションをフローに追加します。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。実行中、コネクションは FHIR サーバー上に新しいリソースを作成します。コネクションは、**id.idPart** フィールドが含まれる **MethodOutcome** リソースを返します。このフィールドには、新しいリソースのリソース ID が含まれ、これをフローの後続のステップにマップできます。

次のステップ

このフローに、新しいリソースに含まれるデータを取得する追加のステップが必要な場合、この FHIR コネクションの前に追加します。これらのステップが追加されたら、新しいリソースを作成する FHIR コネクションの直前にデータマッパーステップを追加します。データマッパーステップで、作成されるリソースのフィールドへマップします。

注意

新しいリソースを作成する FHIR コネクションの直前にデータマッパーステップがないと、コネクションは空のリソースの作成を試みます。空のリソースが作成されるかどうかは、FHIR サーバーの設定によります。

8.5. FHIR サーバー上のリソースにあるすべてのフィールドを更新

インテグレーションでは、FHIR サーバー上のリソースを更新できます。更新された値を提供するには、リソースを更新する FHIR コネクションの直前にデータマッパーステップを追加します。フローの以前のステップは、更新したリソースに含まれるデータを提供することが想定されます。この FHIR コネクションが更新するターゲットリソースのフィールドに、以前のステップのフィールドをマップします。

注意

更新されたリソースには、マップ先のフィールドのみが含まれます。そのため、値が変更するフィールドをマップする他に、リソースに含まれるようにする値の変更しないフィールドをマップする必要があります。特定のリソースフィールドをマップしないと、コネクションは更新されたリソースからそのフィールドを削除します。

以前のステップのリソース ID をこのコネクションのリソース ID にマップするようにしてください。これはコネクションが変更しない唯一のリソースフィールドになります。FHIR サーバーに、更新されるリソースのリソース ID を持つリソースがない場合、コネクションはそのリソース ID で新しいリソース

を作成します。これは、ユーザーが選択したリソース ID で新しいリソースを追加する唯一の方法になります。

リソースを更新または作成するには、FHIR コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- リソースを更新または作成する FHIR サーバー上へのコネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、最後のコネクションを追加するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. FHIR サーバーでリソースの更新に使用する FHIR コネクションをクリックします。
3. **Choose an action** ページで **Update** を選択します。
4. **Resource Type** フィールドをクリックし、FHIR リソースタイプのリストを表示します。更新または作成するリソースのタイプを選択または入力します。
5. 更新または作成するリソースタイプに他のリソースタイプが含まれている場合、**Contained Resource Types** フィールドで含まれているタイプを選択します。複数のタイプを選択する必要がある場合は **Ctrl** キーを使用します。
含まれるリソースタイプを特定すると、データマッパーは含まれるリソースにあるフィールドを表示できるようになります。含まれるリソースタイプを指定しないと、含まれるリソースにあるフィールドをマップ元またはマップ先としてマッピングすることができません。実際に含まれないリソースを選択しても問題はありません。
6. **Next** をクリックして、このコネクションをフローに追加します。
7. 更新されたリソースに含まれるデータを取得する追加のステップがフローに必要な場合は、この FHIR コネクションの前に追加します。
8. インテグレーションビジュアライゼーションで、リソースを更新する FHIR コネクションの前にあるプラス記号をクリックします。
9. **Data Mapper** をクリックします。
10. データマッパーで以下を行います。
 - a. リソース ID をターゲットリソース ID にマップします。これは、新しいリソースのリソース ID を指定する唯一の方法になります。
 - b. 更新されたリソースまたは新しいリソースに含める各リソースフィールドにマップします。値が変更しないフィールドおよび値を更新する必要があるフィールドにマップするようにしてください。
 - c. **Done** をクリックして、データマッパーステップをフローに追加します。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。実行中、コネクションは FHIR サーバーでリソースを更新または作成し、**id.idPart** フィールドが含まれる **MethodOutcome** リソースを返します。このフィールドには更新または作成されたリソースの ID が含まれ、これをフローの後続のステップにマップできます。

8.6. FHIR サーバー上のリソースにある指定のフィールドを更新

インテグレーションでは、FHIR サーバー上にあるリソースの個別のフィールドを更新できます。これには、フローの途中で FHIR コネクションを追加するか、FHIR コネクションをシンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- 更新するリソースを持つ FHIR サーバーへのコネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. リソースの更新に使用する FHIR コネクションをクリックします。
3. **Choose an action** ページで **Patch** を選択します。
4. **Resource Type** フィールドをクリックし、FHIR リソースタイプのリストを表示します。更新するフィールドを持つリソースのタイプを選択または入力します。
5. **Number of Operations** フィールドに、更新するフィールドの数を指定します。
6. フローの以前のステップから値をマップする場合は **Resource id** フィールドを空白のままにしておきます。これが通常の作業になります。その他の場合は、更新するフィールドが含まれるリソースのリソース ID を指定します。
7. 更新が記述された JSON パッチを指定します。JSON パッチは [What is a JSON Patch?](#) に記述されている形式である必要があります。

Patient リソースのパッチ例:

active フィールドを **true** に設定します。

```
[[{"op": "replace", "path": "/active", "value": true}]]
```

患者の名前の配列である **name** フィールドの値を置き換えます。この例では、患者の名前は 1 つです。患者に複数の名前がある場合は、同様の形式のパッチで患者の名前をすべて置き換えることができます。

```
[[{"op": "replace", "path": "/name", "value": [{"given": "Bob"}]}]]
```

患者の名を置き換えます。

```
[[{"op": "replace", "path": "/name/0/given/0", "value": "John"}]]
```

名の前に別の名前を患者に追加します。

```
[[ {"op": "add", "path": "/name/0", "value": {"given": ["Anthony"]} } ]]
```

更新によっては、**Patch** アクションの設定時に JSON パッチを指定する必要があります。たとえば、リソースフィールドの値を別のフィールドにコピーすることがあります。その他の更新では、**JSON Patch** フィールドを空白のままにし、代わりに各更新を定義する値をマップすることができます。

8. **Next** をクリックして、このコネクションをフローに追加します。
9. JSON パッチやリソース ID を指定しなかった場合は、データマッパーステップを追加します。
 - a. フロービジュアライゼーションで、追加したばかりの FHIR コネクションの前にあるプラス記号をクリックします。
 - b. **Data Mapper** をクリックします。
Target パネルには、更新する各フィールド用の番号が付けられたフォルダーが表示されます。**Patch** アクションを設定するとき、更新するフィールドの数を指定します。データマッパーは、この数のフォルダーを **Target** パネルに表示します。たとえば、更新するフィールドの数として 3 を指定すると、**1**、**2**、および **3** のラベルが付いた 3 つのターゲットフォルダーが表示されます。
 - c. 更新する各フィールドをいずれかのターゲットフォルダーのフィールドにマップします。
 - i. **Target** パネルでフォルダーを展開し、**op**、**path**、および **value** の 3 つのフィールドを表示します。
 - ii. ソースフィールド、定数、またはプロパティをターゲット **path** フィールドにマップします。パスは更新するリソースフィールドを識別します。パスの値では、数字はリストフィールドのインデックスを示し、スラッシュは子フィールドに通じます。たとえば、**/name/1/given/1/value** パスをマップし、指定の name フィールドの値を更新します。
 - iii. フィールドの現在の値を置き換えるデフォルトの更新操作を許可するか、ソースフィールド、定数、またはプロパティを **op** フィールドにマップし、フィールドの更新方法を示します。可能な操作の詳細は、[JSON patch operations](#) を参照してください。
 - iv. フィールドを更新して新しい値を取得する場合は、ソースフィールド、定数、またはプロパティをターゲット **value** にマップします。これは、フィールドに含まれるようにする新しい値になります。
 - d. **Patch** アクションの設定時にリソース ID を指定しなかった場合は、リソース ID をターゲット **id** フィールドにマップします。
 - e. **Done** をクリックして、データマッパーステップをフローに追加します。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。実行中、コネクションは指定のリソースフィールドを更新し、**id.idPart** が含まれる **MethodOutcome** リソースを返します。このフィールドには更新されたリソースの ID が含まれます。

8.7. FHIR サーバーで異なるタイプのリソースの作成

インテグレーションでは、FHIR サーバーで 2 つ以上のリソースを作成でき、各リソースを異なるタイプにすることができます。たとえば、新しい患者のリソースと新しい医療担当者のリソースを 1 つのコ

ネクションで FHIR サーバーに追加できます。これには、FHIR コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- リソースを追加する FHIR サーバーへのコネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. 新しいリソースを作成し、FHIR サーバーに追加するために使用する FHIR コネクションをクリックします。
3. **Choose an action** ページで **Transaction** を選択します。
4. **Included Resource Types** フィールドに、このコネクションが作成するリソースタイプを選択します。**Ctrl** キーを使用して複数のリソースタイプを選択します。コネクションは、選択した各タイプのリソースを1つ作成します。
5. **Next** をクリックして、このコネクションをフローに追加します。
6. 新しいリソースにフィールドを入力するには、このコネクションの前にデータマッピングステップを追加します。
 - a. フロービジュアライゼーションで、追加したばかりのコネクションの前にあるプラス記号をクリックします。
 - b. **Data Mapper** をクリックします。
 - c. データマッピングでソースフィールドをターゲットリソースフィールドにマップします。新しいリソースには、マップ先のターゲットフィールドのみが含まれます。
 - d. **Next** をクリックして、データマッピングステップをフローに追加します。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。実行中、コネクションはすべてのリソースを FHIR サーバーに追加します。エラーがある場合は、コネクションはすべてのリソースを FHIR サーバーに追加しません。アクションが無事実行されると、新しい個々のリソースのリソース ID が含まれるトランザクションリソースが返されます。これらのリソース ID をフローの後続のステップにマップできます。

8.8. FHIR サーバーからのリソースの削除

インテグレーションでは、FHIR サーバーからリソースを削除できます。これには、フローの途中に FHIR コネクションを追加するか、FHIR コネクションをシンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- 削除するリソースを持つ FHIR サーバーへの接続が作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後の接続を選択するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。Fuse Online が最後の接続を選択するよう要求する場合は、このステップをスキップします。
2. リソースの削除に使用する FHIR 接続をクリックします。
3. **Choose an action** ページで **Delete** を選択します。
4. **Resource Type** フィールドをクリックし、FHIR リソースタイプのリストを表示します。削除するリソースのタイプを選択または入力します。
5. **Resource Id** フィールドに、削除するリソースのリソース ID を指定します。インテグレーションの初期のステップからリソース ID をマップする予定である場合は、このフィールドを空白のままにしておきます。
6. **Resource version** フィールドには、リソースのバージョン ID を任意で指定します。最新バージョンのリソースを削除する場合や、以前のステップから削除するリソースのバージョン ID をマップする場合は、このフィールドを空白のままにしておきます。
7. **Next** をクリックして、この接続をフローに追加します。

結果

インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。実行中、接続は FHIR サーバーの指定のリソースフィールドを削除し、**id.idPart** フィールドが含まれる **MethodOutcome** リソースを返します。このフィールドには、削除されたリソースのリソース ID が含まれます。

次のステップ

Resource Id フィールドまたは **Resource version** フィールドに値をマップするには、この接続の前にデータマップステップを追加します。

第9章 FTP または SFTP サーバーへの接続

インテグレーションでは、FTP または SFTP サーバーに接続して、ファイルをダウンロードまたはアップロードすることができます。これには、FTP または SFTP コネクションを作成し、インテグレーションフローに追加します。詳細は以下のセクションを参照してください。

- 「FTP または SFTP コネクションの作成」
- 「FTP または SFTP サーバーからのファイルの取得」
- 「FTP または SFTP サーバーへのファイルのアップロード」

9.1. FTP または SFTP コネクションの作成

インテグレーションで FTP または SFTP サーバーからファイルをダウンロードしたり、アップロードするには、その FTP または SFTP サーバーへのコネクションを作成します。同じコネクションを任意の数のインテグレーションに追加できます。

前提条件

接続するサーバーのホスト名を知っている必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクターを表示します。
3. File Transfer Protocol (FTP) を使用するコネクションを作成する場合は、**FTP コネクター** をクリックします。
Secure File Transfer Protocol (SFTP) を使用するコネクションを作成する場合は、**SFTP コネクター** をクリックします。
4. コネクションを設定します。
 - FTP コネクションの場合は以下を行います。
 - 指定するパラメーターは **Host** のみです。接続するサーバーのホスト名を入力します。たとえば、FTP ホストの名前が **FTP.WEST** の場合、**FTP.WEST** をそのまま入力します。**ftp://FTP.WEST** のようにプロトコルは指定しないでください。
 - **Port** は必要なパラメーターで、デフォルト値は **21** になります。これは、FTP サーバーがリッスンするポートになります。
 - その他のパラメーターは、不必要であるかデフォルト値が指定されています。ほとんどのインテグレーションではデフォルトの値が適しています。これらのパラメーターの説明は、この手順の後に記載されています。
 - SFTP コネクションの場合は、以下のパラメーターに値が必要になります。
 - **Host** は、接続する SFTP サーバーのホスト名です。たとえば、SFTP ホストの名前が **SFTP.EAST** の場合、**SFTP.EAST** をそのまま入力します。**sftp://SFTP.EAST** のようにプロトコルを指定しないでください。
 - **Port** のデフォルト値は **22** になります。これは、SFTP サーバーがリッスンするポートになります。

- **User Name** は SFTP サーバーへのアクセスに使用するアカウントのユーザー名です。
 - **Password** はそのユーザーに関連するパスワードです。
 - その他のパラメーターにはデフォルト値が指定されています。ほとんどのインテグレーションではデフォルトの値が適しています。これらのパラメーターの説明は、この手順の後に記載されています。
5. **Validate** をクリックします。Fuse Online は即座に接続の検証を試み、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
 6. 検証に成功したら **Next** をクリックします。
 7. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**XLight FTP Server** を入力します。
 8. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。
 9. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **XLight FTP Server** が表示されます。

その他のパラメーターの説明

- **Connect timeout** のデフォルト値は 10000 ミリ秒で、接続の確立まで最大 10 秒待機することを意味します。10 秒が経過しても接続が確立されなかった場合、Fuse Online は **Reconnect delay** で定義されたミリ秒間待機した後に、再接続を試みます。
- **Reconnect delay** のデフォルト値は 1000 ミリ秒で、再接続を試みる前の待機時間を表します。
- **Maximum reconnect attempts** のデフォルト値は 3 です。Fuse Online は接続の確立を最大 3 回試みます。
- **Binary file transfer mode** がデフォルトで使用されます。転送モードを ASCII にする場合は **No** を選択します。
- **Passive connection mode** のデフォルトは **Yes** で、これが通常の推奨モードになります。パッシブモードでは、ファイアウォールの問題を避けるために、クライアントがサーバーで通信チャンネルを開きます。**No** を選択するとアクティブモードが使用されます。
- **Disconnect from the server after use** のデフォルト値は **No** です。アクションの実行後、接続は維持されます。接続がアップロードまたはダウンロードを実行した後にサーバーから接続を切断する場合は **Yes** を選択します。
- **Data timeout** のデフォルト値は 30000 ミリ秒で、Fuse Online が応答を待つ最大時間を示します。

9.2. FTP または SFTP サーバーからのファイルの取得

FTP または SFTP コネクションが対象のファイルを見つけたときにインテグレーションを実行するには、FTP または SFTP コネクションをインテグレーションの最初の接続として追加します。

前提条件

FTP または SFTP コネクションが作成済みである必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、FTP または SFTP サーバーのポーリングに使用する FTP または SFTP コネクションをクリックします。
4. **Choose an action** ページで **Download** を選択します。
5. ファイルが1つの場合は、**File name expression** フィールドにファイル名に解決される **Apache Camel Simple 言語** 表現を入力します。正規表現を指定することはできません。コネクションはこのファイルを見つけるためにサーバーをポーリング (定期的を確認) し、見つかったらそのファイルをダウンロードします。複数のファイルをダウンロードする場合はこのフィールドを空白のままにしておきます。
6. **FTP directory** フィールドに、ポーリングする server ディレクトリーの絶対または相対パスを入力します。コネクションは、このディレクトリーのコンテンツを確認し、コンテンツを見つければファイルをすべてダウンロードします。
7. **Milliseconds before polling starts** フィールドでは、デフォルトの 1000 ミリ秒を使用するか、値を変更します。
8. **Milliseconds before the next poll** フィールドでは、デフォルトの 500 ミリ秒を使用するか、値をミリ秒単位で変更します。これは、ポーリングの間隔になります。
9. **Delete after download** フィールドでは、デフォルト値の **No** を使用するか、**Yes** を選択してファイルをダウンロードした後にサーバーから削除します。
10. **Next** をクリックして、アクションの出力タイプを指定します。
11. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
12. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
13. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。

このタイプを指定するコネクションを使用する場合は、このタイプを作成または編集するときに

このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときにこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。

14. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
15. **Next** をクリックします。

結果

コネクションは、インテグレーションビジュアライゼーションの最初に表示されます。

9.3. FTP または SFTP サーバーへのファイルのアップロード

インテグレーションでは、フローの途中で FTP または SFTP サーバーへファイルをアップロードしたり、シンプルなインテグレーションを終了することができます。これには、FTP または SFTP コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- FTP または SFTP コネクションが作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. ファイルのアップロードに使用する FTP または SFTP コネクションをクリックします。
3. **Choose an action** ページで **Upload** を選択します。
4. 特定の1つのファイルのみをアップロードする場合は、**File name expression** フィールドにファイル名に解決される [Apache Camel Simple 言語](#) 表現を入力します。これは、アクションによってサーバーにアップロードされるファイルの名前になります。正規表現を指定することはできません。複数のファイルをアップロードする場合はこのフィールドを空白のままにしておきます。
5. **FTP directory** フィールドには server ディレクトリーの絶対または相対名を入力します。**File name expression** フィールドに表現が含まれる場合、コネクションはこのディレクトリーに指定されたファイルを格納します。**File name expression** フィールドが空白である場合、コネクションは前のステップで取得したすべてのファイルをこのディレクトリーにアップロードします。
6. **If file exists** フィールドには、サーバーのファイルと同じパスや名前を持つファイルをアップロードする場合に行う処置を示します。デフォルトの **Override** を使用してサーバー上のファイルをアップロードするファイルでオーバーライドするか、以下の1つを選択します。
 - **Append** は、アップロードするファイルの内容をサーバー上のファイルに追加します。

- **Fail** は、**GenericFileOperationException** を出力します。インテグレーションはエラー状態になりません。
 - **Ignore** はファイルをアップロードしません。インテグレーションは何も問題がないことを前提で実行を継続します。
 - **Move** は1つのファイルの名前を変更します。
 - **TryRename** は一時的な名前でファイルをアップロードし、希望のファイル名に変更します。この操作は、希望の名前を持つファイルの存在を確認しないため、既存ファイルのチェックを行う場合よりもほとんどのサーバーで操作が速くなります。
7. **Temporary file prefix while copying** フィールドには、文字列を指定します。ファイルのアップロード中にこの文字列がファイル名の先頭に追加されます。これにより、コネクションはサーバー上で一時ファイルに書き込みを行い、その一時ファイルの名前を適切な名前に変更することが可能になります。これは、非常に大きなファイルをアップロードする場合にロックを減らすのに便利です。
 8. **Temporary file name while copying** フィールドには、文字列を指定します。ファイルのアップロード中にそのファイルの名前が変更されます。これにより、コネクションはサーバー上で一時ファイルに書き込みを行い、その一時ファイルの名前を適切な名前に変更することが可能になります。これは、非常に大きなファイルをアップロードする場合にロックを減らすのに便利です。
 9. **Next** をクリックしてアクションの入力タイプを指定します。
 10. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
 11. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 12. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 13. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。

14. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。

第10章 GOOGLE アプリケーションへの接続

インテグレーションは、Google アプリケーションである Gmail、Google カレンダー、および Google スプレッドシートに接続できます。詳細は以下のトピックを参照してください。

- [「Fuse Online を Google クライアントアプリケーションとして登録」](#)
- [「Gmail への接続」](#)
- [「Google カレンダーへの接続」](#)
- [「Google スプレッドシートへの接続」](#)

10.1. FUSE ONLINE を GOOGLE クライアントアプリケーションとして登録

インテグレーションで Gmail、Google カレンダー、または Google スプレッドシートへの接続を可能にするには、Fuse Online 環境を Google クライアントアプリケーションとして登録する必要があります。登録中に、Google API を有効にします。さらに、有効にした Google API へアクセスするために Fuse Online が使用するクレデンシャルも作成します。

登録を完了すると、Gmail、Google カレンダー、および Google スプレッドシートへの接続を複数作成することができます。各接続を複数のインテグレーションで使用することができます。Google アプリケーションへの各接続は、登録中に取得する同じ Google クライアント ID と Google クライアントシークレットを使用できますが、各接続はユーザーが選択する別の Google アカウントにアクセスすることもできます。

この手順では、Gmail API、Google Calendar API、および Google Sheet API を有効にしますが、3つの API すべてではなく必要な API のみを有効にすることができます。ただし、3つすべての API ではなく、必要な API のみを有効にします。API を有効にしなかったアプリケーションへの接続を後で作成する必要がある場合は、この手順にしたがって API を有効にすることができます。

重要

Fuse Online 用の新しい Google クライアントアプリケーションを作成する必要があります。Google が新しいクライアントアプリケーションに提供するクレデンシャルには、期限切れのアクセストークンを更新するために使用される更新トークンが含まれています。この更新トークンは、Fuse Online クライアントアプリケーションがクレデンシャルを初めて使用する場合のみ使用できます。Fuse Online では、Gmail、Google カレンダー、および Google スプレッドシートへの接続はすべて同じ Google クライアント ID および Google クライアントシークレットを使用することができます。この場合、更新トークンは Google アプリケーションへのすべての接続に対して使用できます。Fuse Online ユーザーインターフェイスで接続の詳細を表示するときに、**Validate** ボタンをクリックしないでください。検証はクレデンシャルを2回目に使用することになり、更新トークンはクライアントクレデンシャルの一部ではなくなります。ただし、Google アプリケーションに再接続することはできません。



開発環境で、Fuse Online ではない他の OAuth クライアントで使用している Google クライアント ID および Google クライアントシークレットを使用する場合は注意してください。Fuse Online はオフラインのアクセスを必要とし、そのアクセスは最初の OAuth の交換で要求されます。他の OAuth クライアントがすでに OAuth の交換を行い、オフラインのアクセスを要求しなかった場合、Fuse Online は後続の OAuth の交換でオフラインのアクセスを取得することはできません。最初の交換でオフラインアクセスの要求があったことを確認できない場合は、Fuse Online の新しい Google クライアントアプリケーションを作成してください。

前提条件

- Fuse Online を Google クライアントアプリケーションとして登録するために使用する Google アカウントにログインできる必要があります。

手順

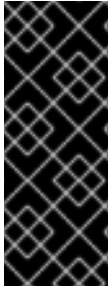
1. Fuse Online で以下を行います。
 - a. 左側のナビゲーションパネルで **Settings** をクリックします。
 - b. **Settings** ページで、上部付近にあるコールバック URL の右側で  をクリックし、Fuse Online 環境のコールバック URL をクリップボードにコピーします。この手順の後半でこの URL が必要になります。
2. 別のブラウザタブで <https://console.developers.google.com> にアクセスし、以下を行います。
 - a. Fuse Online を Google クライアントアプリケーションとして登録するために使用する Google アカウントにログインしていることを確認します。または、別の Google アカウントを選択し、そのアカウントにログインします。
Google API のページの上部に現在の Google プロジェクトの名前が表示されます。
 - b. 現在のプロジェクトを使用して、Fuse Online へ承認を付与するには、次のステップに進みます。別の Google プロジェクトで Fuse Online へ承認を付与するには、そのプロジェクトを選択または作成します。Google アカウントに既存のプロジェクトがない場合は、作成する必要があります。
 - c. Google API を有効にします。
 - i. 左上隅で  をクリックし、**API and Services > Library** を選択します。
 - ii. 必要な場合は下方向にスクロールし、**G Suite** 行のカードを確認します。
 - iii. **Google Calendar API** カードをクリックすると、Google Calendar API が有効になっていることを示すページが表示されます。
 - iv. 下方向にスクロールして **Gmail API** カードをクリックすると、Gmail API が有効になっていることを示すページが表示されます。
 - v. 左上隅で  をクリックし、**API and Services > Library** を選択します。
 - vi. 必要な場合は下方向にスクロールし、**G Suite** 行のカードを確認します。
 - vii. **Google Sheets API** カードをクリックすると、Google Sheets API が有効になっていることを示すページが表示されます。
 - d. クライアントアプリケーションに名前を付けます。
 - i. **MANAGE** をクリックします。
 - ii. 表示されるページの左側のナビゲーションパネルで **Credentials** をクリックします。右側にある **Credentials in APIs & Services** をクリックします。
 - iii. **OAuth consent screen** タブをクリックします。

- iv. 表示されるページの **Application Name** フィールドに Fuse Online クライアントアプリケーションの名前を入力します。たとえば、**Fuse Online client application** を入力します。
 - v. 他のフィールドは無視します。
 - vi. **Save** をクリックします。
 - e. 以下のようにクライアントアプリケーションのクレデンシャルを取得します。
 - i. **Create Credentials** の右側の下矢印をクリックしてメニューを表示し、**OAuth client ID** を選択します。
 - ii. 表示されるページで **Web application** を選択し、その他のコンテンツを表示します。
 - iii. **Name** フィールドに、Fuse Online 環境の OAuth クライアント ID の名前を入力します。これは、クライアントアプリケーション自体に入力した名前とは異なります。たとえば、**OAuth client ID for Fuse Online** を入力します。
 - iv. **Authorized JavaScript origins** は無視します。
 - v. この手順の最初で Fuse Online 環境からコピーしたコールバック URL を **Authorized redirect URIs** フィールドに貼り付けます。
 - vi. **Create** をクリックして、Fuse Online 環境のクライアント ID とシークレットを表示します。
 - f. クライアント ID フィールドの右側で  をクリックし、クライアント ID をクリップボードにコピーします。
3. Fuse Online **Settings** ページに戻り、以下を行います。
 - a. Gmail、Google カレンダー、および Google スプレッドシートのエントリーを展開します。
 - b. 各 Google アプリケーションの **Client ID** フィールドに、コピーした Google クライアント ID を貼り付けます。
4. Google の開発者サイトに戻り、クライアントシークレットフィールドの右側で  をクリックし、クライアントシークレットをクリップボードにコピーします。
5. Fuse Online **Settings** ページに戻り、各 Google アプリケーションのエントリーで以下を行います。
 - a. コピーした Google クライアントシークレットを **Client Secret** フィールドに貼り付けます。
 - b. **Save** をクリックします。**Registration Successful!** という通知が表示されるはずです。
 - c. エントリーを折りたたみます。

結果

有効にした各 Google アプリケーション API に対して、そのアプリケーションへの接続を作成できます。

Fuse Online から Google アプリケーションへのすべてのコネクションは、同じ Google クライアント ID と Google クライアントシークレットを使用します。



重要

Google アプリケーションへのコネクションを持つインテグレーションが継続的に正しく動作するようにするため、Google クライアント ID と Google クライアントシークレットにはトークン更新情報が含まれます。そのため、新しいクレデンシャルを取得しないでください。新しいクレデンシャルを取得すると、各 Google コネクションを再作成し、以前のコネクションを新しいコネクションに置き換え、Google コネクションを使用する各インテグレーションを再パブリッシュする必要があります。

10.2. GMAIL への接続

特定の Gmail アカウントがメールを受け取ったときにインテグレーションの実行をトリガーするには、Gmail コネクションを最初のコネクションとしてシンプルなインテグレーションに追加します。インテグレーションで特定の Gmail アカウントからメールを送信するには、以下のいずれかを行います。

- Gmail コネクションをフローの途中に追加します。
- Gmail コネクションを追加して、シンプルなインテグレーションを完了します。

インテグレーションで Gmail に接続するための一般的なステップは次のとおりです。

1. [Fuse Online を Google クライアントアプリケーションとして登録します](#)。
2. Gmail コネクションを作成します。これには、コネクションがアクセスを許可される Gmail アカウントを選択します。
3. インテグレーションが Gmail アカウントからメールを送信する場合は、送信メールの内容を入力する方法を決定します。
4. インテグレーションフローに Gmail コネクションを追加します。
5. メールを送信する Gmail コネクションでは、任意でインテグレーションデータを電子メールフィールドにマップします。

情報や手順は以下を参照してください。

- [「Gmail コネクションの作成」](#)
- [「送信メールの代替の作成方法」](#)
- [「ポーリングによる Gmail メッセージの返信時にインテグレーションをトリガー」](#)
- [「Gmail アカウントからのメールの送信」](#)

10.2.1. Gmail コネクションの作成

Gmail コネクションを作成する場合、特定の1つの Gmail アカウントにアクセスできるようにコネクションを許可する必要があります。Gmail コネクションの作成後、複数のインテグレーションを追加することができます。

前提条件

- [Fuse Online が Google クライアントアプリケーションとして登録](#) され、Gmail API が有効になっている必要があります。
- Gmail の Fuse Online **Settings** ページエントリには、Fuse Online を Google クライアントアプリケーションとして登録して取得したクライアント ID とクライアントシークレットの値があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックし、利用可能なコネクターを表示します。コネクターは、1つ以上のコネクションを作成するために使用するテンプレートです。
3. **Gmail** コネクターをクリックします。
4. **Configure Connection** ページで **Connect Gmail** をクリックします。 **Sign in with Gmail** ページが表示されます。
Connect Gmail が表示されない場合、Fuse Online 環境は Google クライアントアプリケーションとして登録されていません。[Fuse Online を Google クライアントアプリケーションとして登録](#) を参照してください。Fuse Online 環境が Google クライアントアプリケーションとして登録されていない状態で Gmail コネクションを作成しようとすると、Fuse Online に承認情報の入力を要求する複数のフィールドが表示されます。これらのフィールドに値を入力して Gmail コネクションを作成することは可能ですが、推奨されません。
5. Google へのログインを要求するページで、このコネクションが Fuse Online からアクセスする Google アカウントのメールアドレスを入力し、**Next** をクリックします。
6. **openshiftapps.com wants to access your Google Account** に対して **Allow** をクリックし、Fuse Online に戻ります。
7. Fuse Online の **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Gmail Connect 1** を入力します。
8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample Gmail connection that uses jkim Gmail account credentials.** を入力します。
9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Gmail Connect 1** が表示されます。

10.2.2. 送信メールの代替の作成方法

シンプルなインテグレーションを完了する Gmail コネクションや、フローの途中にある Gmail コネクションは、アクセスが許可されている Gmail アカウントからメールを送信します。Gmail コネクションが送信するメールの内容を入力する方法は複数あります。メールを送信する Gmail を追加する前に、メールの内容を入力する方法を検討してください。

送信メールの内容を入力する代替方法は次のとおりです。

- メールを送信する Gmail コネクションの直前にデータマッパーステップを追加します。このデータマッパーステップでは、以前のステップから出力されたデータフィールドを Gmail コネクションの **Send Email** アクションフィールドにマップします。 **Send Email** アクションフィールドは次のとおりです。

- Email to
- Email subject
- Email text
- Email cc
- Email bcc

データマッピングステップを追加した場合、**Send Email** アクションフィールドを1つ、一部、またはすべてマップできます。

- Gmail コネクションをフローに追加するとき、**Send Email** アクションフィールドに値を指定してアクションを設定します。値は1つ、一部、またはすべてのフィールドに指定できます。
- データマッピングステップと **Send Email** アクション設定を両方使用して、フィールドに値を入力します。つまり、1つ以上のフィールドを指定して **Send Email** アクションを設定でき、その他のフィールドに値を入力するデータマッピングステップを追加することもできます。
Send Email アクションフィールドに直接指定する値は、**Send Email** アクションフィールドにマップされる値よりも優先されます。つまり、コネクションの追加時およびアクションの設定時に値を指定して **Send Email** フィールドに値を入力し、さらに同じフィールドに値をマッピングした場合に、アクション設定で指定した値は常にマップされた値よりも優先されます。たとえば、**Email to** アクションフィールドに **people@redhat.com** を指定し、前のステップのメールフィールドも Gmail の **Email to** フィールドにマップする場合、インテグレーションは常に **people@redhat.com** をメールアドレスとして使用します。

メールを送信する Gmail コネクションを追加するとき、すべてのアクション設定パラメーターは任意となります。これは、インテグレーションデータを **Send Email** アクションフィールドにマップして、メールの内容を完全に入力することを選択する場合があるからです。しかし、アクション設定での指定またはマッピングのいずれかを行って、**Email to** フィールドにメールアドレスが存在する必要があります。メッセージ送信先のメールアドレスがないと、Fuse Online はランタイムエラーを生成し、インテグレーションの実行が停止します。

10.2.3. ポーリングによる Gmail メッセージの返信時にインテグレーションをトリガー

特定の Gmail アカウントが受信したメールを基にインテグレーションを実行するには、Gmail コネクションをシンプルなインテグレーションの最初のコネクションとして追加します。インテグレーションの実行時、Gmail コネクションはユーザーが制御する間隔毎に、このアカウントでメールをチェックします。コネクションが未読のメールを見つけると、そのメールをインテグレーションの次のステップに渡し、デフォルトではメールに **read** (既読) マークを付けます。

前提条件

- メールを取得する Gmail アカウントへのアクセスが許可される Gmail コネクションが作成されている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する Gmail コネクションをクリックします。

4. **Choose an action** ページで **Receive Email** アクションを選択します。
5. **Receive Email** アクションを設定します。
 - a. **Delay** フィールドには、デフォルトの 30 秒を使用するか、インテグレーションが新規メールをチェックする頻度を指定します。
 - b. 未読のメールをすべて取得する場合は **Labels** フィールドを空白のままにしておきます。特定のメールのみを取得する場合は、コネクションがアクセスしている Gmail アカウントで使用されるラベルのコンマ区切りリストを指定します。デフォルトでは、インテグレーションはこれらのラベルが付けられた未読のメールを取得します。
 - c. コネクションが同じメールを 2 度返さないようにするため、**Mark as read** を選択します。**Mark as read** を選択しないと、メールが未読または既読であるかに関わらず、コネクションはアカウントの受信トレイにあるメールを返します。1 つ以上のラベルを指定した場合は、メールが未読または既読であるかに関わらず、指定したラベルが付けられたメールを返します。
 - d. **Max Results** フィールドには、デフォルトの 5 を使用するか、各ポーリングに対してコネクションが返すことができるメールの最大数を指定します。コネクションがこの値を超える数の未読メールを見つけた場合、最も新しい **Max Results** メールを返します。Gmail コネクションが 1 つ以上のメールを返すとき、インテグレーションはメールをバッチとして処理します。つまり、Fuse Online はバッチに対してインテグレーションを 1 度実行します。
 - e. **Next** をクリックして、この Gmail コネクションを最初のコネクションとしてインテグレーションに追加します。コネクションはインテグレーションフローの最初のステップとして表示されます。

10.2.4. Gmail アカウントからのメールの送信

インテグレーションでは、フローの途中で Gmail アカウントからメールを送信したり、シンプルなインテグレーションを終了するために Gmail アカウントからメールを送信することができます。これには、Gmail コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Gmail コネクションが作成されている必要があります。
- [送信するメールの内容を入力する代替方法](#) について熟知し、このようなメールの内容を入力する計画がある必要があります。
- Fuse Online はインテグレーションの追加を要求したり、シンプルなインテグレーションの最後のコネクションを選択するよう要求します。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. メール送信に使用する Gmail コネクションをクリックします。
3. **Choose an action** ページで **Send Email** をします。

4. Configure Send Email ページで以下の1つを行います。

- この接続の直前にデータマッパーステップを追加する計画があり、インテグレーションデータを **Send Email** フィールドにマップする計画がある場合は、すべてのフィールドを空白にしておきます。
- 次の **Send Email** フィールドの1つ以上に情報を入力します。複数のメールアドレスはコマンドで区切ります。
 - a. **Email To**
 - b. **Email Subject**
 - c. **Email Text**
 - d. **Email cc**
 - e. **Email bcc**
- 一部のフィールドを空白のままにし、他の一部のフィールドに情報を入力します。この Gmail 接続の直前にデータマッパーステップを追加する計画があり、インテグレーションデータを一部のメールにマップする場合、アクション設定でこれらのフィールドを空白にします。必要に応じて他のアクションフィールドに値を入力します。

Send Email アクション設定フィールドに指定した値は、以前のステップからマップした値よりも優先されます。

5. **Next** をクリックして、接続をフローに追加します。

結果

インテグレーションフローで追加した場所に接続が表示されます。

次のステップ

この接続の前にデータマッパーステップを追加する予定がある場合は最初に、追加する予定のある他の接続をこのフローに追加します。その後、データマッパーステップを追加します。

10.3. GOOGLE カレンダーへの接続

ポーリングが更新を Google カレンダーに戻すときにインテグレーションの実行をトリガーするには、Google カレンダー接続を最初の接続としてシンプルなインテグレーションに追加します。イベントをカレンダーに追加するか、カレンダーのイベントを更新するには、以下のいずれかを行います。

- Google カレンダーをフローの途中に追加します。
- Google カレンダーを追加して、シンプルなインテグレーションを完了します。

Google カレンダーへの接続に関する詳細は、以下を参照してください。

- [「Google カレンダー接続の作成」](#)
- [「ポーリングが Google カレンダーからイベントを返すときにインテグレーションをトリガー」](#)
- [「Google カレンダーからの特定イベントの取得」](#)

- 「[Google カレンダーでのイベントの更新](#)」
- 「[イベントの Google カレンダーへの追加](#)」

10.3.1. Google カレンダーコネクションの作成

Google カレンダーのコネクションを作成するとき、1つの特定の Google アカウントに関連する Google カレンダーへアクセスするためコネクションを承認します。Google カレンダーコネクションの作成後、複数のインテグレーションに追加できます。

前提条件

- [Fuse Online](#) が [Google クライアントアプリケーションとして登録](#) され、Google Calendar API が有効になっている必要があります。
- Google カレンダーの Fuse Online **Settings** ページエントリーには、Fuse Online を Google クライアントアプリケーションとして登録して取得したクライアント ID とクライアントシークレットの値があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックし、利用可能なコネクタを表示します。コネクタは、1つ以上のコネクションを作成するために使用するテンプレートです。
3. **Google Calendar** コネクタをクリックします。
4. **Configure Connection** ページで **Connect Google Calendar** をクリックします。Google のログインページが表示されます。
Connect Google Calendar が表示されない場合、Fuse Online 環境は Google Calendar API が有効な状態で Google クライアントアプリケーションとして登録されていません。[Fuse Online を Google クライアントアプリケーションとして登録](#) を参照してください。Fuse Online 環境が Google クライアントアプリケーションとして登録されていない状態で Google カレンダーコネクションを作成しようとすると、Fuse Online に承認情報の入力を要求する複数のフィールドが表示されます。これらのフィールドに値を入力して Google カレンダーコネクションを作成することは可能ですが、推奨されません。
5. Google のログインページで、このコネクションが Fuse Online からアクセスする Google アカウントを選択し、**Next** をクリックします。
6. **openshiftapps.com wants to access your Google Account** プロンプトに対して **Allow** をクリックし、Fuse Online に戻ります。
7. Fuse Online の **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Google Calendar Work Connection** を入力します。
8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample Google Calendar connection that uses my Google work account.** を入力します。
9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Google Calendar Work Connection** が表示されます。

10.3.2. ポーリングが Google カレンダーからイベントを返すときにインテグレーションをトリガー

指定した Google カレンダーからイベントを取得したときにインテグレーションの実行をトリガーするには、Google カレンダーコネクションを最初のコネクションとしてシンプルなインテグレーションに追加します。インテグレーションの実行時、Google カレンダーコネクションはユーザーが制御できる間隔で Google カレンダーのイベントをチェックします。コネクションが、Google カレンダーの **Get Events** アクションの設定方法に準拠するイベントを見つけると、コネクションはイベントをインテグレーションの次のステップに渡します。

Google カレンダーコネクションが 2 つ以上のイベントを返すとき、Fuse Online は返された各イベントに対してインテグレーションを実行します。たとえば、ポーリングによって 5 つのイベントが返されると、Fuse Online はインテグレーションを 5 回実行します。

前提条件

- Google カレンダーコネクションが作成済みである必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する Google カレンダーコネクションをクリックします。
4. **Choose an action** ページで **Get Events** アクションを選択します。
5. **Get Events** アクションを設定します。
 - a. **Delay** フィールドには、デフォルトの 30 秒を使用するか、インテグレーションがカレンダーをチェックする頻度を指定します。
 - b. **Calendar name** フィールドには、この Google カレンダーコネクションによるアクセスが承認されている Google アカウントからアクセス可能なカレンダーの名前を入力します。
 - c. **Max Results** フィールドでは、デフォルトの 5 を使用するか、各ポーリングに対して返すことができるコネクションの最大イベント数を指定します。
 - d. 以下の 1 つを選択し、コネクションが使用するイベントのチェックの開始日を示します。コネクションはこの日付と後続の日付をチェックします。
 - **Consume from the current date ahead** を選択すると、現在の日付が開始日になります。たとえば、このオプションを選択し、**Max Results** が 5 の場合、Google カレンダーコネクションはイベントのチェックは現在の日付から開始され、見つかったイベントを最初から最大 5 つ返します。最後のポーリング後にイベントが更新または追加されたかは関係ありません。
 - **Consume from the last event update date on the next poll** を選択すると、前回のポーリングで返された最新の更新イベントの日付が開始日になります。たとえば、このオプションを選択し、**Max Results** が 5 の場合、Google カレンダーコネクションは最大 5 つの更新または新規イベントを返します。コネクションが新規または送信イベントをチェックする開始日は、前回のポーリングで返された最新の更新イベントの日付になります。

- e. 任意で、**Query for events** フィールドにテキストを指定し、ポーリングが返すことができるイベントをフィルターすることができます。接続は、最低でも1つのイベントフィールドに指定されたテキストが含まれるイベントのみを返します。
たとえば、クエリーフィールドに **Standup meeting** を指定するとします。この場合、ポーリングはイベントフィールドに **Standup meeting** が含まれるイベントのみを返します。
6. **Next** をクリックし、この Google カレンダー接続をインテグレーションの最初の接続として追加します。

結果

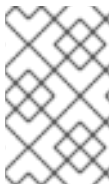
接続はシンプルなインテグレーションの最初のステップとして表示されます。

10.3.3. Google カレンダーからの特定イベントの取得

インテグレーションでは、フローの途中で特定の Google カレンダーイベントを取得できます。特定イベントの取得は、以下のような場合に便利です。

- 後続の Google カレンダー接続でイベントを更新する場合。
- 後続の Twitter 接続を使用してイベントを発表する場合。

1つのイベントを取得するには、Google カレンダーをフローの途中に追加します。



注記

本リリースでは、シンプルなインテグレーションの最後の接続で特定のイベントを取得することはサポートされますが、これは特に便利なことではありません。これは今後のリリースで変更される予定です。

前提条件

- 取得するイベントがある Google カレンダーへのアクセスが承認されている Google カレンダー接続が作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。
2. 接続するカレンダーへのアクセスが承認されている Google カレンダー接続をクリックします。
3. **Choose an action** ページで **Get a Specific Event** を選択します。
4. 取得するイベントがあるカレンダーの名前を指定します。
5. 取得するイベントの ID を指定します。イベント ID を取得するには、以下の1つを行います。
 - 以前の Google カレンダー接続からマップします。
 - 以下のようにカレンダーから手動で取得します。
 - a. ブラウザーで、取得するイベントが含まれるカレンダーを表示します。

- b. URL の末尾に **?gsessionid=OK&eventdeb=1** を追加し、カレンダーを再表示します。
 - c. カレンダーで、取得するイベントをクリックします。
 - d. イベントのポップアップで、 をクリックし、**Troubleshooting info** を選択します。
 - e. Google カレンダーが表示するポップアップで、**eid=** に続く文字列をコピーします。たとえば、イベント ID は **p1pva2a4t504gbsha12di9ch6k_20181107T150000Z*** のようになります。
6. **Next** をクリックして、接続をフローに追加します。

結果

接続はフローで追加した場所に表示されます。

次のステップ

以前のステップの値をこの接続のフィールドにマップする場合は、データマッパーステップを追加します。最初に、必要なすべての接続をフローに追加します。次に、この接続の直前にデータマッパーステップを追加します。

10.3.4. イベントの Google カレンダーへの追加

インテグレーションでは、フローの途中で Google カレンダーにイベントを追加したり、シンプルなインテグレーションを終了するために Google カレンダーにイベントを追加することができます。これには、Google カレンダー接続をフローの途中で追加するか、シンプルなインテグレーションの最後の接続として追加します。

前提条件

- イベントを追加する Google カレンダーへのアクセスが承認されている Google カレンダー接続が作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後の接続を選択するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。Fuse Online が最後の接続を選択するよう要求する場合は、このステップをスキップします。
2. イベントを追加するカレンダーへのアクセスが承認されている Google カレンダー接続をクリックします。
3. **Choose an action** ページで **Create Event** アクションを選択します。
4. **Create Event** アクションを設定します。
 - a. 新しいイベントのタイトルを入力します。
 - b. イベントの追加先となる Google カレンダーの名前を入力します。
 - c. その他の入力パラメーターにはデータを入力できますが、以前の Google カレンダー接続で取得したイベントから値をマップすることも可能です。

5. **Next** をクリックして、コネクションをフローに追加します。

結果

インテグレーションフローで追加した場所にコネクションが表示されます。

次のステップ

値をこの Google カレンダーコネクションのフィールドにマップする場合は、フローにデータマッパーステップを追加します。最初に、必要なすべてのコネクションをフローに追加します。次に、この Google カレンダーコネクションの直前にデータマッパーステップを追加します。

10.3.5. Google カレンダーでのイベントの更新

インテグレーションでは、フローの途中で Google カレンダーのイベントを更新したり、シンプルなインテグレーションを終了するために Google カレンダーのイベントを更新することができます。これには、Google カレンダーコネクションをフローの途中で追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。



重要

本リリースでは、**Update Event** アクションには各イベントフィールドに値が必要です。よって、ほぼすべての場合で、[更新するイベントを取得する](#) Google カレンダーコネクションを追加し、次にイベントを更新する Google カレンダーコネクションを追加してから、この2つの Google カレンダーコネクションの間にデータマッパーステップを挿入する必要があります。

前提条件

- 更新するイベントがある Google カレンダーへのアクセスが承認されている Google カレンダーコネクションが作成済みである必要があります。
- フローには、Google カレンダーへの以前のコネクションがあり、そのコネクションが更新するイベントを取得します。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. 更新するイベントがあるカレンダーへのアクセスが承認されている Google カレンダーコネクションをクリックします。
3. **Choose an action** ページで **Update Event** を選択します。
4. **Update Event** アクションを設定します。
 - a. 更新するイベントのタイトルを入力します。
 - b. 更新する各イベントフィールドの内容を入力します。フィールドの内容を変更したくない場合は、イベントフィールドに内容を入力しないでください。

5. **Next** をクリックして、コネクションをフローに追加します。インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。
6. インテグレーションビジュアライゼーションで、追加したばかりのコネクションの前にあるプラス記号をクリックします。
7. **Data Mapper** をクリックします。
8. データマッパーで、更新されたイベントの変更のなかった各イベントフィールドを、イベントを取得した Google カレンダーコネクションから、イベントを更新した Google カレンダーコネクションの対応フィールドにマップします。
更新しているフィールドをマップしないでください。**Update Event** アクションの設定時にフィールドをマップし、そのフィールドの更新も指定すると、Fuse Online はマップされた値を使用します。
9. 右上の **Done** をクリックし、データマッパーステップを追加します。

10.4. GOOGLE スプレッドシートへの接続

Google スプレッドシートコネクションがスプレッドシートデータまたはスプレッドシートプロパティを返すときにインテグレーションの実行をトリガーするには、Google スプレッドシートコネクションを最初のコネクションとしてシンプルなインテグレーションに追加します。スプレッドシートの値またはプロパティを更新するか、チャートまたはピボットテーブルをスプレッドシートに挿入して、シンプルなインテグレーションを終了するには、Google スプレッドシートコネクションを最後のコネクションとしてシンプルなインテグレーションに追加します。

フローの途中で、スプレッドシートで値を取得、作成、または更新する Google スプレッドシートコネクションを追加でき、スプレッドシートにチャートやピボットテーブルを挿入する Google スプレッドシートコネクションを追加することもできます。

Google スプレッドシートへの接続に関する詳細は、以下を参照してください。

- [「Google スプレッドシートコネクションの作成」](#)
- [「スプレッドシートデータの取得によるインテグレーションのトリガーまたはフローの途中でスプレッドシートデータを取得」](#)
- [「ポーリングがスプレッドシートプロパティを返すときにインテグレーションをトリガー」](#)
- [「スプレッドシートの作成」](#)
- [「シートのデータの更新」](#)
- [「データをシートに追加」](#)
- [「スプレッドシートプロパティの更新」](#)
- [「チャートをシートに追加」](#)
- [「ピボットテーブルをシートに追加」](#)

10.4.1. Google スプレッドシートコネクションの作成

Google スプレッドシートのコネクションを作成するとき、1つの特定の Google アカウント (ユーザーが選択する) に関連する Google スプレッドシートへアクセスするためにコネクションを承認します。Google スプレッドシートコネクションの作成後、複数のインテグレーションに追加できます。

前提条件

- [Fuse Online](#) が [Google クライアントアプリケーション](#) として登録され、Google Sheets API が有効になっている必要があります。
- Google スプレッドシートの Fuse Online **Settings** ページエントリーには、Fuse Online を Google クライアントアプリケーションとして登録して取得したクライアント ID とクライアントシークレットの値があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックし、利用可能なコネクタを表示します。コネクタは、1つ以上のコネクションを作成するために使用するテンプレートです。
3. **Google Sheets** コネクタをクリックします。
4. **Configure Connection** ページで **Connect Google Sheets** をクリックします。Google のログインページが表示されます。
Connect Google Sheets が表示されない場合、Fuse Online 環境は Google Sheets API が有効な状態で Google クライアントアプリケーションとして登録されていません。[Fuse Online を Google クライアントアプリケーションとして登録](#) を参照してください。Fuse Online 環境が Google クライアントアプリケーションとして登録されていない状態で Google スプレッドシートコネクションを作成しようとすると、Fuse Online に承認情報の入力を要求する複数のフィールドが表示されます。これらのフィールドに値を入力して Google スプレッドシートコネクションを作成することは可能ですが、推奨されません。
5. Google のログインページで、このコネクションが Fuse Online からアクセスする Google アカウントを選択し、**Next** をクリックします。
6. **openshiftapps.com wants to access your Google Account** プロンプトに対して **Allow** をクリックし、Fuse Online に戻ります。
7. Fuse Online の **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Google Sheets Work Connection** を入力します。
8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample Google Sheets connection that uses my Google work account.** を入力します。
9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Google Sheets Work Connection** が表示されます。

10.4.2. スプレッドシートデータの取得によるインテグレーションのトリガーまたはフローの途中でスプレッドシートデータを取得

Google スプレッドシートからデータを取得したときにインテグレーションの実行をトリガーするには、Google スプレッドシートコネクションを最初のコネクションとしてシンプルなインテグレーションに追加します。インテグレーションの実行時、Google スプレッドシートは指定された間隔でスプレッドシートをポーリングして特定したデータを取得し、データをインテグレーションの次のステップに渡します。

フローの途中でスプレッドシートデータを取得するには、Google スプレッドシートコネクションを途

中のコネクションとして追加します。実行中、Fuse Online はこのコネクションの処理を開始すると同時に指定したデータのスプレッドシートをポーリングします。つまり、コネクションは間隔が経過するのを待たずにスプレッドシートをポーリングします。

スプレッドシートの特定のシートからデータを取得するには、Google スプレッドシートコネクションのアクションを設定するときシート名を指定します。1つの特定のコネクションは1つのシートからのみデータを取得できます。

ポーリングの合間に、コネクションが返すように設定されているシート値に変更がない場合は、次のポーリングは前回のポーリングと同じ値を返します。

前提条件

- データの取得元となるスプレッドシートへのアクセスが承認されている Google スプレッドシートコネクションが作成されている必要があります。
- Google スプレッドシートコネクションがインテグレーションの実行をトリガーする場合はインテグレーションを作成します。また、Fuse Online は最初のコネクションの選択を要求します。
- Google スプレッドシートコネクションがフローの途中にある場合は、インテグレーションにすでに最初と最後のコネクションがあります。Fuse Online は、ステップを選択してインテグレーションを追加するよう要求します。

手順

1. 使用する Google スプレッドシートコネクションをクリックします。
2. **Choose an action** ページで、最初のコネクションに **Get sheet values** アクションを選択するか、途中のコネクションに **Retrieve sheet values** アクションを選択します。
3. アクションを設定するには以下を行います。
 - a. **SpreadsheetId** フィールドには、この Google スプレッドシートコネクションによるアクセスが承認されている Google アカウントからアクセス可能な Google のスプレッドシートの ID を入力します。
スプレッドシート ID を取得するには、ブラウザでスプレッドシートを表示します。ID は **d/** と **/edit** の間にある URL の一部です。
 - b. **Range** フィールドに、取得するデータを識別する Google A1 表記を入力します。デフォルトは **A:A** です。
たとえば、4つの列が含まれる1つのシートがあるスプレッドシートからすべてのデータを取得する場合、**A:D** を指定します。行 5 から 15 からデータを取得する場合は **A5:D15** を指定します。

データの取得元となるスプレッドシートに複数のシートがある場合は、スプレッドシート名とともに、開始セルの座標と終了セルの座標を指定します。複数のシートがある場合にシート名を指定しないと、コネクションはスプレッドシートの最初のシートからデータを取得します。たとえば、**2019!A1:D5** を指定すると、**2019** という名前のシートからデータを取得します。そのシートで列 A から列 D の行 1 から 5 のデータを取得します。
 - c. **Major dimension** フィールドで、デフォルトの **Rows** を使用するか、**Columns** を選択します。
Rows は、行オブジェクトのコレクションを返すようアクションを設定します。各行オブジェクトには該当する各列の値が含まれます。**Major dimension** が **Rows** である場合、Fuse Online は **A**、**B**、**C** などの見出しではなく、意味のある列見出しをフィールド名としてデータマッパーで表示します。

Columns は、列オブジェクトのコレクションを返すようアクションを設定します。各列オブジェクトには該当する各行の値が含まれます。

- d. **Header row number** フィールドで **Major dimension** が **Rows** に設定された場合、コネクションが取得するデータの列見出しが含まれる行の番号を任意で入力します。ヘッダ行を指定すると、Fuse Online でスプレッドシートから見出しを取得できるようになります。ヘッダ行を指定しないと、列見出しのデフォルトは、コネクションが取得するデータの範囲にある各列のレター見出しになります。後続のページで取得した見出しやレター見出しを編集できます。



注記

Google スプレッドシートの途中のコネクションを設定する場合、残りの設定オプションは必要なく、Fuse Online は残りの設定を要求しません。ステップ 4 に進みます。

- e. **Split results** フィールドでデフォルトの **No** を許可するか、**Yes** を選択します。**No** を設定すると、データを値のコレクションとして返すアクションを設定します。これにより、コネクションは行オブジェクトのコレクションまたは列オブジェクトのコレクションをフローの次のステップに渡します。**Yes** を選択すると、**Major dimension** の設定にしたがってコネクションは返されたデータを分割できます。たとえば、**Major dimension** が **Rows** に設定されている場合、コネクションは行オブジェクトを返します。各行オブジェクトはフローの実行を個別にトリガーします。Fuse Online は返された各行オブジェクトに対してフローを 1 度実行します。たとえば、ポーリングによって 5 つの行が返されると、Fuse Online はフローを 5 回実行します。

Fuse Online は、フローに追加できる個別の分割ステップ (split step) および集約ステップ (aggregate step) も提供します。1 つ以上のステップで個別のオブジェクトを処理し、行または列オブジェクトを集約する場合、Google スプレッドシートコネクションの結果を分割しないでください。代わりに、デフォルトの **No** を使用し、このコネクションの後にフローに分割ステップ (split step) を追加します。フローに集約ステップ (aggregate step) を使用する場合は分割ステップ (split step) が必要になります。

- f. **Delay** フィールドには、デフォルトの 30 秒を使用するか、コネクションがスプレッドシートデータを取得する頻度を指定します。
- g. ポーリングが返すことができる行や列の数を制限したくない場合は **Max results** フィールドにデフォルトの 0 を指定します。**Max results** の設定は、結果メトリクスのメジャーディメンションの設定に適用されます。メジャーディメンションにコネクションが返すデータを制限するには、整数を指定します。たとえば、メジャーディメンションが列で、その **Max results** が 25 に設定された場合、ポーリングによって 25 行以下の値が返されます。

Range でメジャーディメンションを指定し、**Max results** も指定する場合、ポーリングはより小さい数値を使用して返すデータの量を判断します。そのため、**Max results** の値を設定することは、**Range** のマイナーディメンションのみを指定する場合に便利です。たとえば、スプレッドシートに 30 行あり、各行に 3 つの値の列がある例を考えてみましょう。**Major dimension** が **Rows** で **Range** が **A:C** であるとします。**Max results** を 10 に設定した場合、ポーリングは 10 行のデータを返します。**Max results** の設定に 0 を指定すると、ポーリングは 30 行を返します。しかし、**Range** を **A1:C15** として指定し、さらに **Max results** を 25 として指定した場合、ポーリングは 15 行を返します。**Range** を **A1:C30** として指定し、さらに **Max results** を 25 として指定した場合、ポーリングは 25 行を返します。

4. **Next** をクリックして、**Major dimension** が **Rows** に設定されたときに接続が取得するデータの列名を表示します。**Major dimension** が **Columns** の場合はこのフィールドの内容は無視され、**Next** をクリックしてこの手順を完了できます。
Column names に表示される値は、データマッピングステップが表示するフィールド名になります。ヘッダ行番号を指定した場合、Fuse Online はデータ取得元のスプレッドシートの行から見出しを表示します。ヘッダ行番号フィールドを空白のままにした場合は、取得するデータの範囲にある各列に対して1つの文字 (**A**、**B**、**C** など) を表示します。
5. 任意で、データマッピングステップで表示するフィールド名が含まれるように **Column names** フィールドを編集します。このフィールドには、**Name,Address,City,State,Zip** のように、空白のないコンマ区切りリストを含める必要があります。
6. **Next** をクリックして、この Google スプレッドシート接続をフローに追加します。

次のステップ

Google スプレッドシート接続を最初のコネクションとして追加した場合、Fuse Online はインテグレーションの最後のコネクションを追加するよう要求します。最初と最後のコネクションをインテグレーションに追加したら、インテグレーションに必要な他のコネクションを追加します。

スプレッドシート値を取得する接続の後に、データマッピングステップを追加します。データマッピングでは、スプレッドシートの値を取得するアクションの設定に応じてソースフィールドが表示されます。メジャーディメンションが **Rows** である場合、データマッピングはターゲットにマップできるフィールドとして列名を表示します。メジャーディメンションが **Columns** である場合は、データマッピングはターゲットにマップできるフィールドとして行インデックスを表示します。

その他のリソース

スプレッドシートでセルのグループを指定する [Google A1 表記](#)

10.4.3. ポーリングがスプレッドシートプロパティを返すときにインテグレーションをトリガー

Google スプレッドシートからプロパティを取得したときにインテグレーションの実行をトリガーするには、Google スプレッドシート接続を最初のコネクションとしてシンプルなインテグレーションに追加します。インテグレーションの実行時、Google スプレッドシートは指定された間隔でスプレッドシートをポーリングしてスプレッドシートプロパティを取得し、結果をインテグレーションの次のステップに渡します。

プロパティには、スプレッドシートのタイトル、ロケール、およびタイムゾーンが含まれます。ポーリングの合間にシートプロパティに変更がない場合は、次のポーリングは前回のポーリングと同じ値を返します。

前提条件

- プロパティの取得元となるスプレッドシートへのアクセスが承認されている Google スプレッドシート接続が作成されている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションの開始に使用する Google スプレッドシート接続をクリックします。

4. **Choose an action** ページで **Get spreadsheet properties** アクションを選択します。
5. **Get spreadsheet properties** アクションを設定します。
 - a. **SpreadsheetId** フィールドには、この Google スプレッドシートコネクションによるアクセスが承認されている Google アカウントからアクセス可能な Google のスプレッドシートの ID を入力します。
スプレッドシート ID を取得するには、ブラウザでスプレッドシートを表示します。ID は **d/** と **/edit** の間にある URL の一部です。
 - b. **Delay** フィールドには、デフォルトの 30 秒を使用するか、コネクションがプロパティを取得する頻度を指定します。
6. **Next** をクリックし、この Google スプレッドシートコネクションをインテグレーションの最初のコネクションとして追加します。

結果

インテグレーションに最初のコネクションが追加され、Fuse Online でインテグレーションの最後のコネクションが選択されるよう要求されます。

10.4.4. スプレッドシートの作成

フローの途中に新しいスプレッドシートを作成するには、Google スプレッドシートコネクションを最初のコネクションと最後のコネクションの間に追加します。スプレッドシートを作成する Google スプレッドシートコネクションでシンプルなインテグレーションを終了することはできません。同じコネクションの新しいスプレッドシートにデータを追加することはできません。そのため、スプレッドシートを作成し、同じフローのスプレッドシートにデータを追加する場合、フローには 2 つの Google スプレッドシートコネクションが必要になります。1 つのコネクションはスプレッドシートを作成し、後続のコネクションはデータをスプレッドシートに追加します。

前提条件

- スプレッドシートを作成する Google アカウントへのアクセスが承認される Google スプレッドシートコネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. スプレッドシートの作成に使用する Google アカウントへのアクセスが承認される Google スプレッドシートコネクションをクリックします。
3. **Choose an action** ページで **Create spreadsheet** を選択します。
4. **Create spreadsheet** アクションを設定します。
 - a. **Title** フィールドに新しいスプレッドシートのタイトルを入力します。
 - b. **Time Zone** フィールドに新しいスプレッドシートが使用するタイムゾーンを入力します (例: **India Standard Time**、**Tokyo**)。

- c. **Locale** フィールドに新しいスプレッドシートのロケールを入力します (例: **Canada**、**Hong Kong**)。
5. **Next** をクリックして、接続をフローに追加します。インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。

結果

スプレッドシートを作成する接続によって新しいスプレッドシートのスプレッドシート ID が返されます。後続のステップで、このスプレッドシート ID を更新などのために新しいスプレッドシートにアクセスする接続のスプレッドシート ID にマップできます。

10.4.5. シートのデータの更新

インテグレーションでは、フローの途中でスプレッドシートのデータを更新したり、シンプルなインテグレーションを終了するためにスプレッドシートでデータを更新することができます。これには、Google スプレッドシート接続をフローの途中に追加するか、シンプルなインテグレーションの最後の接続として追加します。

前提条件

- 更新する Google スプレッドシートへのアクセスが承認されている Google スプレッドシート接続が作成されている必要があります。
- 更新するスプレッドシートのスプレッドシート ID にアクセスできる必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後の接続を選択するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。Fuse Online が最後の接続を選択するよう要求する場合は、このステップをスキップします。
2. 更新するスプレッドシートへのアクセスが承認されている Google スプレッドシート接続をクリックします。
3. **Choose an action** ページで **Update sheet values** を選択します。
4. **Update sheet values** を設定します。
 - a. **SpreadsheetId** フィールドで以下の1つを行います。
 - 更新するスプレッドシートの ID を入力します。
 - フィールドは空白のままにします。この場合、この接続をフローに追加した後に、この接続の前にデータマッピングステップを追加する必要があります。データマッピングステップでは、スプレッドシート ID を以前の接続からこの接続にマップします。以前の接続はこの接続で更新するスプレッドシートにアクセスする必要があります。
 - b. **Range** フィールドでは、Google A1 表記を使用して更新するスプレッドシートのセルを指定します。開始および終了座標を指定します。たとえば、**A1:D4** はスプレッドシートの最初のシートで最初の 4 行の最初の 4 列を更新します。デフォルトは **A:A** で、スプレッドシートの最初のシートで最初の列を更新します。

- c. **Major dimension** フィールドで、デフォルトの **Rows** を使用するか、**Columns** を選択します。**Rows** は、行オブジェクトを使用してシートを更新するアクションを設定します。各行オブジェクトには、更新する各列の値が含まれます。**Columns** は、列オブジェクトを使用してシートを更新するアクションを設定します。各列オブジェクトには、更新する各行の値が含まれます。
 - d. **Value input option** フィールドには、シートを更新するために受信するデータを Google スプレッドシートが解釈する方法を示します。デフォルトの **Unspecified** を使用すると、Google スプレッドシートは認識するデータを自動的に変換します。たとえば、入力データが日付である場合、Google スプレッドシートは日付としてフォーマットします。入力データが 10 進数の場合は、Google スプレッドシートは 10 進数としてフォーマットします。
 - **Unspecified** のデフォルトは Google Sheets API 設定で、このデフォルトは **User entered** になります。
 - **Raw** を選択すると、Google スプレッドシートは入力データをそのまま挿入します。
 - **User entered** は、認識可能なデータの自動変換を有効にします。
 - e. **Next** をクリックして、コネクションをフローに追加します。インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。
5. フローに他のコネクションを追加する場合はここで追加し、これらの手順に戻ります。
 6. フローにすべてのコネクションを追加したら、インテグレーションビジュアライゼーションで、シート値を更新する Google スプレッドシートコネクションの直前にあるプラスマークをクリックします。
 7. **Data Mapper** カードをクリックして、データマッピングステップをフローに追加します。
 8. データマッパーで以下を行います。
 - a. **Update sheet values** アクションを設定したときにスプレッドシート ID を指定しなかった場合、ソース **spreadsheetId** をターゲット **spreadsheetId** にマップします。
 - b. フィールドをソースステップからターゲットスプレッドシートにマッピングして、シートの更新に使用するデータを指定します。たとえば、フィールドを別のスプレッドシートやデータベースからマッピングする場合があります。
必要な場合、シートの値を更新する Google スプレッドシートコネクションを編集し、**Range** または **Major dimension** の設定を変更することができます。これらの設定を変更すると、データマッパーは変更に応じて異なるターゲットフィールドを表示します。
 - c. 右上の **Done** をクリックし、データマッパーステップを追加します。

10.4.6. データをシートに追加

フローの途中でシートにデータを追加したり、シンプルなインテグレーションを終了するためにシートにデータを追加することができます。これには、Google スプレッドシートコネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- データを追加する Google スプレッドシートへのアクセスが承認される Google スプレッドシートコネクションが作成されている必要があります。
- データを追加するスプレッドシートの ID にアクセスできる必要があります。

- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後の接続を選択するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。Fuse Online が最後の接続を選択するよう要求する場合は、このステップをスキップします。
2. データを追加するスプレッドシートへのアクセスが承認される Google スプレッドシート接続をクリックします。
3. **Choose an action** ページで **Append values to a sheet** を選択します。
4. **Append values to a sheet** アクションを設定します。
 - a. **SpreadsheetId** フィールドで以下の1つを行います。
 - 値を追加するスプレッドシートの ID を入力します。
 - フィールドは空白のままにします。この場合、この接続をフローに追加した後に、この接続の前にデータマッピングステップを追加する必要があります。データマッピングステップでは、スプレッドシート ID を以前の接続からこの接続にマップします。以前の接続はこの接続でデータを追加するスプレッドシートにアクセスする必要があります。
 - b. **Range** フィールドに Google A1 表記を使用し、データを追加するスプレッドシートの範囲を指定します。開始および終了座標を指定します。たとえば、**A3:D3** の場合は最初のシートの 4 列の 3 行目の後にデータを追加します。デフォルトは **A:A** で、スプレッドシートの最初のシートで最初の列にデータを追加します。
接続はデータを上書きすることはありません。接続は指定した範囲の後にデータの追加を開始し、常にデータを適切な場所にある内容に追加します。
 - c. **Major dimension** フィールドで、デフォルトの **Rows** を使用するか、**Columns** を選択します。**Rows** は、行オブジェクトを使用してデータを追加するアクションを設定します。各行オブジェクトには、データを追加する各列の値が含まれます。**Columns** は、列オブジェクトを使用してデータを追加するアクションを設定します。各列オブジェクトには、追加する各行の値が含まれます。
 - d. **Value input option** フィールドには、シートに追加するために受信するデータを Google スプレッドシートが解釈する方法を示します。デフォルトの **Unspecified** を使用すると、Google スプレッドシートは認識するデータを自動的に変換します。たとえば、入力データが日付である場合、Google スプレッドシートは日付としてフォーマットします。入力データが 10 進数の場合は、Google スプレッドシートは 10 進数としてフォーマットします。
 - **Unspecified** のデフォルトは Google Sheets API 設定で、このデフォルトは **User entered** になります。
 - **Raw** を選択すると何もしません。Google スプレッドシートは入力データをそのまま挿入します。
 - **User entered** は、認識可能なデータの自動変換を有効にします。
 - e. **Next** をクリックして、接続をフローに追加します。インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。

Append values to a sheet アクションを設定するために指定した値は、このコネクションをマッピング先またはマッピング元とするためにデータマッパーが表示するフィールドを判断します。

5. フローに他のコネクションを追加する場合はここで追加し、これらの手順に戻ります。
6. フローにすべてのコネクションを追加したら、インテグレーションビジュアライゼーションで、データをシートに追加する Google スプレッドシートコネクションの直前にあるプラスマークをクリックします。
7. **Data Mapper** をクリックして、データマッピングステップをフローに追加します。
8. データマッパーで以下を行います。
 - a. **Append values to a spreadsheet** アクションを設定したときにスプレッドシート ID を指定しなかった場合、ソース **spreadsheetId** をターゲット **spreadsheetId** にマップします。
 - b. フィールドをソースステップからターゲットスプレッドシートにマッピングして、シートに追加するデータを指定します。たとえば、フィールドを別のスプレッドシートやデータベースからマッピングする場合があります。
必要な場合、シートの値を追加する Google スプレッドシートコネクションを編集し、**Range** または **Major dimension** の設定を変更することができます。これらの設定を変更すると、データマッパーは変更に応じて異なるターゲットフィールドを表示します。
 - c. 右上の **Done** をクリックし、データマッパーステップを追加します。

10.4.7. スプレッドシートプロパティーの更新

インテグレーションでは、フローの途中でスプレッドシートのプロパティーを更新したり、シンプルなインテグレーションを終了するためにスプレッドシートでデータを更新することができます。これには、Google スプレッドシートコネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

プロパティーには、スプレッドシートのタイトル、ロケール、およびタイムゾーンが含まれます。

Fuse Online が **Update spreadsheet properties** アクションの設定を要求したら、一部またはすべてのフィールドを空白のままにしておきます。フィールドを空白のままにした場合、このコネクションの前のフローのデータマッパーステップ (後で追加) で、フィールドを以前のステップから空白の **Update spreadsheet properties** アクション設定フィールドにマップします。

前提条件

- プロパティーを更新する Google スプレッドシートへのアクセスが承認されている Google スプレッドシートコネクションが作成されている必要があります。
- プロパティーを更新するスプレッドシートのスプレッドシート ID にアクセスできる必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。

2. プロパティを更新するスプレッドシートへのアクセスが承認されている Google スプレッドシートコネクションをクリックします。
3. **Choose an action** ページで、**Update spreadsheet properties** を選択します。
4. **Update spreadsheet properties** アクションを設定します。
 - a. **SpreadsheetId** フィールドで以下の1つを行います。
 - プロパティを更新するスプレッドシートの ID を入力します。
 - フィールドは空白のままにします。この場合、このコネクションをフローに追加した後、このコネクションの前にデータマッピングステップを追加する必要があります。データマッピングステップでは、スプレッドシート ID を以前のコネクションからこのコネクションにマップします。以前のコネクションはこのコネクションで更新するスプレッドシートにアクセスする必要があります。
 - b. 別のフィールドには、プロパティを変更する場合のみ値を入力します。また、フィールドを空白のままにすることもできます。空白にする場合は、このコネクションの直前に後で追加するデータマッピングステップで、変更するフィールドをマップできます。
 - c. **Next** をクリックして、コネクションをフローに追加します。インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。
5. フローに他のコネクションを追加する場合はここで追加し、これらの手順に戻ります。
6. フローにコネクションがすべて追加されたら、データマッピングステップを使用してスプレッドシートプロパティを更新することができます。
 - a. インテグレーションビジュアライゼーションで、プロパティを更新する Google スプレッドシートコネクションの直前にあるプラス記号をクリックします。
 - b. **Data Mapper** をクリックして、データマッピングステップをフローに追加します。
 - c. データマッピングで、**Update spreadsheet properties** アクションを設定したときにスプレッドシートを指定しなかった場合は、ソース **spreadsheetId** をターゲット **spreadsheetId** にマップします。
 - d. **Update spreadsheet properties** アクションを設定したときに更新された値を指定しなかった、変更する各スプレッドシートプロパティに対し、フィールドをソースステップから変更するターゲットスプレッドシートプロパティにマップします。
 - e. 右上の **Done** をクリックし、データマッピングステップを追加します。

10.4.8. チャートをシートに追加

フローの途中で基本的なチャートや円グラフを Google のスプレッドシートに追加できます。これには、Google スプレッドシートコネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。その後、データマッピングステップをコネクションの直前に追加します。データマッピングステップでは、チャートの場所、プロパティ、および内容を判断するオプションを設定します。

Fuse Online が **Add charts** アクションの設定を要求したら、一部またはすべてのフィールドを空白のままにすることができます。フィールドを空白のままにする場合、このコネクションの直前に追加するデータマッピングステップで値をそのフィールドにマップできます。

前提条件

- チャートを追加する Google スプレッドシートへのアクセスが承認されている Google スプレッドシートコネクションが作成されている必要があります。
- チャートを追加するスプレッドシートの ID にアクセスできる必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。Fuse Online で、インテグレーションを追加するよう要求されるか、シンプルなインテグレーションの最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. チャートを追加するスプレッドシートへのアクセスが承認されている Google スプレッドシートコネクションをクリックします。
3. **Choose an action** ページで **Add charts** を選択します。
4. **Add charts** アクションを設定します。
 - a. **SpreadsheetId** フィールドで以下の1つを行います。
 - チャートを追加するスプレッドシートの ID を入力します。
 - フィールドは空白のままにします。このコネクションの直前のデータマッピングステップで、スプレッドシート ID を以前のコネクションからこのコネクションにマップします。以前のコネクションは、このコネクションでチャートを追加するスプレッドシートにアクセスする必要があります。
 - b. **Chart Title** フィールドにタイトルを入力します。
このフィールドを空白のままにすることもできます。このコネクションの直前のデータマッピングステップで、チャートのタイトルをソースフィールドまたはプロパティから **title** ターゲットフィールドにマップします。
 - c. **Subtitle** フィールドにサブタイトルを入力します。
このフィールドを空白のままにすることもできます。このコネクションの直前のデータマッピングステップで、チャートのサブタイトルをソースフィールドまたはプロパティから **subtitle** ターゲットフィールドにマップします。
5. **Next** をクリックして、コネクションをフローに追加します。インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。
6. フローに他のコネクションを追加する場合はここで追加します。その後、これらの手順に戻ります。
7. インテグレーションビジュアライゼーションで、チャートを追加するコネクションの直前にあるプラス記号をクリックします。
8. **Data Mapper** をクリックします。
データマッピングで、チャートを追加するターゲット Google スプレッドシートコネクションにソースプロパティ、ソースフィールド、またはソース定数をマップします。以下の手順は、各ターゲットフィールドにマップする値が含まれるソースフィールド、ソースプロパティ、またはソース定数があることを仮定します。

必要な値のソースがない場合、**Properties** フォルダーの右にあるプラス記号をクリックして、

データマッパーでソースプロパティを作成します。**Create Property** ダイアログで、正しいターゲットフィールドへのマッピングを容易にする名前をプロパティに付けます。たとえば、**overlayPosition** をプロパティ名とすることができます。プロパティの値には、コネクションがチャートの作成に使用する値を指定します。チャートのオーバーレイポジションフィールドにはセルを特定する A1 表記が必要です。これは、開始ポジションを指定しますが、終了ポジションを指定する必要はなく、終了ポジションがある場合は無視されます。たとえば、**overlayPosition** ソースプロパティの値として **D4** を指定します。新しいプロパティを保存します。

Add charts アクションを設定したときに、スプレッドシート ID、チャートのタイトル、およびチャートのサブタイトルの値を指定した場合は、このフィールドに値をマップしないでください。

9. チャートをシートに追加するには、以下のようにソースプロパティ、ソースフィールド、またはソース定数をターゲットフィールドにマップします。

- a. セルを識別する A1 表記を **overlayPosition** ターゲットフィールドにマップして、チャートの場所を設定します。終了ポジションは必要なく、指定があっても無視されます。マップされた値は、Google スプレッドシートがチャートを配置するセルを特定します。このセルはチャート表示の左上隅になります。

overlayPosition フィールドに値をマップせず、チャートを追加するシートを特定する **sheetId** フィールドにも値をマップしないと、Google スプレッドシートが新しいシートを作成し、チャートを A1 に配置するのがデフォルトの動作になります。

sheetId フィールドに値をマップする場合は、**overlayPosition** フィールドに値をマップする必要があります。

- b. ソース **spreadsheetId** フィールドをターゲット **spreadsheetId** フィールドにマップします。コネクションはこのスプレッドシートのソースデータを使用して、チャートを同じスプレッドシートのシートに追加します。ソースデータとチャートは同じスプレッドシートの異なるシートに配置することができますが、異なるスプレッドシートに配置することはできません。
- c. ターゲット **title** フィールドにマッピングして、チャートタイトルを設定します。コネクションはこの文字列をチャートのタイトルとして挿入します。
- d. ターゲット **subtitle** フィールドにマッピングして、チャートのサブタイトルを設定します。コネクションはこの文字列をチャートのサブタイトルとして挿入します。
- e. 整数を **sheetId** フィールドにマップして、チャートを追加するシートを特定します。**sheetId** フィールドに値をマップする場合、**overlayPosition** フィールドに値をマップする必要もあります。

sheetId フィールドに値をマップせず、**overlayPosition** フィールドにも値をマップしない場合、コネクションは新しいシートを作成し、チャートをその新しいシートに追加します。

sheetId フィールドに値をマップせずに、**overlayPosition** フィールドに値をマップした場合、コネクションは **sourceSheetId** フィールドによって特定されたシートにチャートを配置します。このシートはチャートのデータを提供します。

0 はスプレッドシートの最初のシートを特定します。後続のシートは、シートをブラウザで見るときの URL の最後の部分がシート ID となります。たとえば、以下の URL の最後には **gid=206589330** があります。

<https://docs.google.com/spreadsheets/d/1pa...ngQbKkM/edit#gid=206589330>

これは、シート ID が **206589330** であることを示しています。これが `sheetId` にマップする値になります。

- f. 整数のシート ID をターゲット `sourceSheetId` フィールドにマップして、チャートのデータを提供するシートを特定します。デフォルトは **0** で、スプレッドシートの最初のシートを特定します。
基本のチャートまたは円グラフを追加できます。追加するチャートの種類によって、以下の手順の1つにしたがいます。
- g. 基本のチャートを追加するには、ターゲットパネルで `basicChart` フォルダーを展開し、ソースフィールドを `basicChart` のターゲットフィールドにマップします。
 - i. ターゲット `type` フィールドにマップして、チャートの種類 (**BAR**、**LINE**、**AREA**、または **COLUMN**) を設定します。デフォルトは **COLUMN** です。
 - ii. ターゲット `axisTitleBottom` フィールドにマップして、チャートの横軸のタイトルを設定します。
 - iii. ターゲット `axisTitleLeft` フィールドにマップして、チャートの縦軸のタイトルを設定します。
 - iv. ターゲット `domainRange` フィールドにマップして、チャートが表す情報の上位カテゴリを設定します。このフィールドは Google スプレッドシートの A1 表記を使用します。たとえば、ソースシートの最初の列が **A2** から **A6** までの5つの製品のリストを提供し、チャートが各製品の一部のデータを示すこととします。この場合、この列の最初のセル (**A1**) には **Products** のようなラベルが含まれる必要があります。`domainRange` の設定は **A1:A6** になります。チャートには **A2** から **A6** までのデータが含まれるようになります。チャートにはラベルは表示されません。
 - v. ターゲット `dataRange` フィールドをマップして、チャートが示すソースデータの場所を特定します。このフィールドは Google スプレッドシートの A1 表記を使用します。チャートは1系列のデータのみを提供します。`domainRange` の例を引き続き使用し、最初の列にリストされた各5製品の販売数をソースシートの2番目の列に表し、このデータをチャートに示すこととします。この場合、この列の最初のセル (**B1**) には **Number Sold** のようなラベルが含まれる必要があります。`dataRange` の設定は **B1:B6** になります。チャートには **B2** から **B6** までのデータが含まれます。チャートにはラベルは表示されません。
- h. 円グラフを追加するには、ターゲットパネルで `pieChart` フォルダーを展開し、ソースフィールドを `pieChart` フォルダーのターゲットフィールドにマップします。
 - i. ターゲット `legendPosition` フィールドにマップして、円グラフの凡例の場所を設定します。デフォルトは **LEFT_LEGEND** です。値は **BOTTOM_LEGEND**、**LEFT_LEGEND**、**RIGHT_LEGEND**、**TOP_LEGEND**、または **NO_LEGEND** である必要があります。
 - ii. ターゲット `domainRange` フィールドにマップして、円グラフが表す情報の上位カテゴリを設定します。このフィールドは Google スプレッドシートの A1 表記を使用します。たとえば、ソースシートの最初の列が **A2** から **A6** までの5つの製品のリストを提供し、チャートが各製品の一部のデータを示すこととします。この場合、この列の最初のセル (**A1**) には **Products** のようなラベルが含まれる必要があります。`domainRange` の設定は **A1:A6** になります。チャートには **A2** から **A6** までのデータが含まれるようになります。チャートにはラベルは表示されません。
 - iii. ターゲット `dataRange` フィールドをマップして、チャートが示すソースデータの場所を特定します。このフィールドは Google スプレッドシートの A1 表記を使用します。チャートは1系列のデータのみを提供します。`domainRange` の例を引き続き使用し、

最初の列にリストされた各 5 製品の販売数をソースシートの 2 番目の列に表し、このデータをチャートに示すとしてします。この場合、この列の最初のセル (B1) には **Number Sold** のようなラベルが含まれる必要があります。dataRange の設定は B1:B6 になります。チャートには B2 から B6 までのデータが含まれます。チャートにはラベルは表示されません。

- i. 右上の **Done** をクリックし、データマッピングステップを追加します。

10.4.9. ピボットテーブルをシートに追加

フローの途中でピボットテーブルを Google のスプレッドシートに追加できます。また、シンプルなインテグレーションを完了するために追加することもできます。ピボットテーブルではスプレッドシートデータを集計または並べ替えでき、スプレッドシートデータに関数を適用することもできます。ピボットテーブルを追加するには、Google スプレッドシートコネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。その後、データマッピングステップをコネクションの前に追加します。データマッピングステップでは、ピボットテーブルの場所、プロパティ、および内容を判断するオプションを設定します。

本リリースでは、**Add pivot table** アクションは 1 つの値グループ、1 つの行ピボットグループ、および 1 つの列ピボットグループの定義に限定されます。単一のアクションでの複数グループのサポートは、今後のリリースで追加される予定です。回避策として、同じソーススプレッドシートを基にしてピボットテーブルを追加する複数の Google スプレッドシートコネクションを追加することができます。

前提条件

- ピボットテーブルを追加する Google スプレッドシートへのアクセスが承認されている Google スプレッドシートコネクションが作成されている必要があります。
- ピボットテーブルのソースデータが含まれるスプレッドシートの ID にアクセスできる必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。Fuse Online で、インテグレーションを追加するよう要求されるか、シンプルなインテグレーションの最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. ピボットテーブルを追加するスプレッドシートへのアクセスが承認されている Google スプレッドシートコネクションをクリックします。
3. **Choose an action** ページで **Add pivot tables** を選択します。
4. **Add pivot tables** アクションを設定するには、**SpreadsheetId** フィールドで以下の 1 つを行います。
 - ピボットテーブルを追加するスプレッドシートの ID を入力します。
 - フィールドは空白のままにします。このコネクションの直前に追加するデータマッピングステップで、スプレッドシート ID を以前のコネクションからこのコネクションにマップします。以前のコネクションは、このコネクションでピボットテーブルを追加するスプレッドシートにアクセスする必要があります。

5. **Next** をクリックして、コネクションをフローに追加します。インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。
6. このフローに他のコネクションを追加する場合はここで追加します。その後、これらの手順に戻ります。
7. インテグレーションビジュアライゼーションで、ピボットテーブルをシートに追加するコネクションの直前にあるプラス記号をクリックします。

8. **Data Mapper** をクリックします。

データマッパーで、ピボットテーブルを追加するをターゲット Google スプレッドシートコネクションにソースプロパティ、ソースフィールド、またはソース定数をマップします。以下の手順は、各ターゲットフィールドにマップする値が含まれるソースフィールド、ソースプロパティ、またはソース定数があることを仮定します。

必要な値のソースがない場合、**Properties** フォルダーの右にあるプラス記号をクリックして、データマッパーでソースプロパティを作成します。**Create Property** ダイアログで、正しいターゲットフィールドへのマッピングを容易にする名前をプロパティに付けます。たとえば、**valueLayout** をプロパティ名とすることができます。ピボットテーブルを作成するためにコネクションが使用する値を指定します。ピボットテーブルの値レイアウトフィールドには、**HORIZONTAL** または **VERTICAL** の値が必要であるため、**VERTICAL** を **valueLayout** プロパティの値として指定することがあります。新しいプロパティを保存します。

9. データマッパーで、以下のように新しいピボットテーブルを設定します。

- a. ソース **spreadsheetId** フィールドをターゲット **spreadsheetId** フィールドにマップします。コネクションはこのスプレッドシートのソースデータを使用して、ピボットテーブルを同じスプレッドシートのシートに追加します。ソースデータとピボットテーブルは同じスプレッドシートの異なるシートに配置することができますが、異なるスプレッドシートに配置することはできません。
- b. 整数を **sheetId** フィールドにマップして、ピボットテーブルを追加するシートを特定します。デフォルトは **0** で、スプレッドシートの最初のシートを特定します。後続のシートは、シートをブラウザで見るときに URL の最後の部分がシート ID となります。たとえば、以下の URL の最後には **gid=206589330** があります。
<https://docs.google.com/spreadsheets/d/1pa...ngQbKkM/edit#gid=206589330>
これは、シート ID が **206589330** であることを示しています。これが **sheetId** にマップする値になります。
- c. 整数のシート ID をターゲット **sourceSheetId** フィールドにマップして、ピボットテーブルを生成するためのデータを提供するシートを特定します。**sourceSheetId** に値をマップしないと、コネクションは **sheetId** の値を使用します。**sheetId** にマップされた値がない場合は、**0** を使用します。**0** はスプレッドシートの最初のシートを特定します。
- d. A1 表記の値をターゲット **sourceRange** フィールドにマップして、ピボットテーブルの生成にソースデータの範囲を設定します。たとえば、**A2:D5** を設定します。
- e. ターゲット **valueLayout** フィールドにマップして、ピボットテーブルで使用される値のレイアウトを設定します。値は **VERTICAL** またはデフォルトである **HORIZONTAL** である必要があります。**HORIZONTAL** を指定すると値は列としてレイアウトされます。
- f. A1 表記のセル座標をターゲット **start** フィールドにマップして、ピボットテーブルの左上のセルを設定します。コネクションがピボットテーブルをソースデータが含まれるシートに追加するとき、値を **start** フィールドにマップしないと、デフォルトではピボットテーブ

ルの左上のセルがソースデータ範囲の最初の行にあるセルとなり、ソースデータ範囲の後の最初の空白列にあるセルとなります。ソースデータがあるシートとピボットテーブルがあるシートが異なる場合、デフォルトは A1 です。

- g. 値グループを追加するには、ターゲットパネルで **valueGroups** 展開し、**valueGroups** 以下で値をフィールドにマップします。
 - i. 値グループ名をターゲット **name** フィールドにマップします。
 - ii. 値グループに適用する関数をターゲット **function** フィールドにマップします。値は **SUM**、**COUNT**、**AVERAGE**、**MAX**、**MIN**、または **CUSTOM** である必要があります。デフォルトは **SUM** です。公式を定義するときに **CUSTOM** を **function** にマップします。
 - iii. 値 **CUSTOM** を **function** フィールドにマップした場合、値グループに適用するカスタムの公式をターゲット **formula** フィールドにマップします。
 - iv. 値グループを構築する座標の列名をターゲット **sourceColumn** フィールドにマップします。
- h. 行ピボットグループを追加するには、ターゲットパネルで **columnGroups** 展開し、**columnGroups** 以下で値をフィールドにマップします。
 - i. 行ピボットグループ名をターゲット **label** フィールドにマップします。
 - ii. 行ピボットグループに適用する並べ替え順序をターゲット **sortOrder** フィールドにマップします。値は **ASCENDING** または **DESCENDING** である必要があります。デフォルトは **ASCENDING** です。
 - iii. **true** または **false** をターゲット **showTotals** フィールドにマップします。デフォルトは **true** で、行ピボットグループの合計を表示できます。
 - iv. 行ピボットグループを構築する座標の列名をターゲット **sourceColumn** フィールドにマップします。
- i. 列ピボットグループを追加するには、ターゲットパネルで **rowGroups** 展開し、**rowGroups** 以下で値をフィールドにマップします。
 - i. 列ピボットグループ名をターゲット **label** フィールドにマップします。
 - ii. 列ピボットグループに適用する並べ替え順序をターゲット **sortOrder** フィールドにマップします。値は **ASCENDING** または **DESCENDING** である必要があります。デフォルトは **ASCENDING** です。
 - iii. **true** または **false** をターゲット **showTotals** フィールドにマップします。デフォルトは **true** で、列ピボットグループの合計を表示できます。
 - iv. 列ピボットグループを構築する座標の列名をターゲット **sourceColumn** フィールドにマップします。
- j. 右上の **Done** をクリックし、データマップステップを追加します。

その他のリソース

[ピボットテーブルに関する開発者向け Google ドキュメント](#)

第11章 HTTP および HTTPS エンドポイントへの接続

インテグレーションでは、HTTP または HTTPS エンドポイントへ接続し、**GET**、**PUT**、**POST**、**DELETE**、**HEAD**、**OPTIONS**、**TRACE**、または **PATCH** メソッドを実行できます。これを行うには、HTTP または HTTPS コネクションを作成し、インテグレーションフローに追加します。詳細は以下のセクションを参照してください。

- [「HTTP または HTTPS エンドポイントへのコネクションの作成」](#)
- [「HTTP または HTTPS コネクションのインテグレーションへの追加」](#)

11.1. HTTP または HTTPS エンドポイントへのコネクションの作成

インテグレーションで HTTP の **GET**、**PUT**、**POST**、**DELETE**、**HEAD**、**OPTIONS**、**TRACE**、または **PATCH** メソッドを実行するには、HTTP または HTTPS エンドポイントへのコネクションを作成します。作成後、コネクションを1つ以上のインテグレーションに追加できます。

前提条件

接続するエンドポイントの URL を知っている必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクタを表示します。
3. HTTP (Hyper Text Transfer Protocol) を使用してエンドポイントに接続する場合は、**HTTP** コネクタを使用します。HTTPS (Hyper Text Transfer Protocol Secure) を使用する場合は、**HTTPS** コネクタをクリックします。
4. **Base URL** フィールドにエンドポイントパスを入力します。例: **www.mycompany.com/sales**
5. **Validate** をクリックします。Fuse Online は即座にコネクションを検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、指定したベース URL を訂正し、再度検証を行います。
6. 検証に成功した場合は **Next** をクリックします。
7. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**HTTPS My Company Sales** を入力します。
8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。
9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **HTTPS My Company Sales** が表示されます。

11.2. HTTP または HTTPS コネクションのインテグレーションへの追加

HTTP または HTTPS コネクションを任意の数のインテグレーションに追加できます。

前提条件

- HTTP または HTTPS コネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。Fuse Online はインテグレーションの追加を要求します。または、Fuse Online は最初のコネクションまたは最後のコネクションの選択を要求します。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. フローに追加する HTTP または HTTPS コネクションをクリックします。
3. コネクションが実行するアクションを選択します。
 - 最初のコネクションを追加する場合、使用できるアクションは **Periodic invoke URL** のみになります。このアクションは、指定する間隔でエンドポイントを呼び出し、エンドポイントがデータを返すとインテグレーションをトリガーします。
 - 途中または最後のコネクションを追加する場合、使用できるアクションは **Invoke URL** のみです。このアクションはエンドポイントを一度呼び出します。
4. **URL Path** フィールドに、呼び出すエンドポイントの場所を指定します。
5. **HTTP Method** フィールドでコネクションが実行するメソッドを選択します。デフォルトのメソッドは **GET** です。
 - **GET** は、URL パスでコンテンツを取得します。
 - **PUT** は、URL パスでコンテンツをインテグレーションデータに置き換えます。
 - **POST** は、URL パスのインテグレーションデータを格納し、新しいコンテンツを作成します。
 - **DELETE** は URL パスのコンテンツを削除します。
 - **HEAD** は、URL パスでコンテンツに関するメタデータを取得します。
 - **OPTIONS** は、URL パスで通信オプション設定を取得します。
 - **TRACE** は、テストおよび診断の目的で情報を取得します。
 - **PATCH** は、インテグレーションデータに応じて URL パスでコンテンツを部分的に更新します。
6. URL を定期的呼び出す最初のコネクションを追加する場合、**Period** フィールドにデフォルトの間隔である 1 秒を使用するか、数値と単位 (ミリ秒、秒、分、または時間) を指定して呼び出しの間の待ち時間を示します。
7. **Next** をクリックし、アクションの入力または出力タイプを指定します。
8. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。

- **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
9. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 10. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 11. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 12. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

第12章 IRC への接続

IRC (Internet Relay Chat) は、IRC クライアントを使って誰でもリアルタイムでテキストメッセージを送受信できるプロトコルです。インテグレーションでは、IRC への接続は以下のいずれかを行うことができます。

- 指定した IRC ニックネームがプライベートメッセージを受信したときにシンプルなインテグレーションの実行をトリガーします。インテグレーションはメッセージをフローの次のステップに渡します。たとえば、見込み顧客の名前を受信する IRC ニックネームを指定することができます。インテグレーションはメッセージを受信すると Salesforce に接続してリードを作成することができます。
- 特定のチャンネルへメッセージを配信します。メッセージは指定するニックネームから配信されます。たとえば、インテグレーションが FTP サーバーからファイルをダウンロードし、何らかの方法で処理するとき便利です。インテグレーションフローはメッセージを特定のチャンネルに送信し、プロセスの成功を示すことができます。

IRC をインテグレーションに接続するには、IRC コネクションを作成し、コネクションをインテグレーションフローに追加します。詳細は以下を参照してください。

- [「IRC コネクションの作成」](#)
- [「メッセージの受信時にインテグレーションの実行をトリガーするために IRC コネクションを追加」](#)
- [「IRC コネクションを追加してメッセージを送信」](#)

12.1. IRC コネクションの作成

インテグレーションでは、IRC コネクションは指定した IRC ニックネームに送信されたメッセージを取得したり、チャンネルでニックネームにメッセージを送信することができます。任意の数のインテグレーションで同じ IRC コネクションを使用できます。

前提条件

- 接続先の IRC サーバーのホスト名およびポートを知っている必要があります。
- IRC サーバーにパスワードが必要な場合は、そのパスワードを知っている必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクターを表示します。
3. IRC コネクターをクリックします。
4. **Hostname** フィールドに IRC ホスト名を入力する必要があります。IP アドレスまたは名前 (例: **chat.freenode.net**) を指定することができます。
5. **Port** フィールドに IRC サーバーがリスンしているポート (例: **6665**) を入力します。
6. IRC サーバーにパスワードが必要な場合は **Password** フィールドにそのパスワードを入力します。

7. **Validate** をクリックします。Fuse Online は即座に接続を検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、接続設定値を修正し、再度検証を行います。
8. 検証に成功した場合は **Next** をクリックします。
9. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**IRC Freenode** を入力します。
10. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。
11. **Save** をクリックし、作成したばかりの接続を含む、利用可能な接続の更新済みリストを表示します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **IRC Freenode** が表示されます。この接続を任意の数の接続に追加できます。

12.2. メッセージの受信時にインテグレーションの実行をトリガーするために IRC 接続を追加

指定した IRC ニックネームに送信されたプライベートメッセージを接続が発見すると、インテグレーションを開始する IRC 接続はインテグレーションの実行をトリガーします。

前提条件

- IRC 接続が作成されている必要があります。
- メッセージの送受信するための IRC ニックネームが必要です。
- シンプルなインテグレーションを作成することになります。Fuse Online はインテグレーションの開始方法の選択を要求します。

手順

1. インテグレーションの開始に使用する IRC 接続をクリックします。
2. **IRC Receive Private Message** アクションを選択します。
3. **Channels** フィールドに任意で1つ以上の IRC チャンネルをコンマで区切って指定します。
4. **Nickname** フィールドには、インテグレーションが使用する IRC ニックネームを指定します。このニックネームは指定されたすべてのチャンネルに参加します。実行中、インテグレーションはこのニックネームに送信されたメッセージを取得し、インテグレーションの次のステップに渡します。
5. 指定のニックネームにパスワードがある場合は **NickServ Password** フィールドにパスワードを入力します。
6. **Next** をクリックして、アクションの出力タイプを指定します。
7. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。

- **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
8. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 9. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 10. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 11. **Next** をクリックして、コネクションをフローに追加します。

結果

コネクションはシンプルなインテグレーションの最初のコネクションで、Fuse Online はインテグレーションの最後のコネクションを選択するよう要求します。

12.3. IRC コネクションを追加してメッセージを送信

インテグレーションでは、フローの途中でメッセージを1つ以上の IRC チャンネルに送信したり、シンプルなインテグレーションを終了するためにメッセージを1つ以上の IRC チャンネルに送信することができます。これには、IRC コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- IRC コネクションが作成されている必要があります。
- メッセージの送受信するための IRC ニックネームが必要です。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. フローに追加する IRC コネクションをクリックします。
3. **IRC Send Message to Channel** を選択して、メッセージを1つ以上のチャンネルに送信します。

- **Channels** フィールドに、コンマで区切られた1つ以上の IRC チャンネルを入力します。実行中、このコネクションは指定した IRC ニックネームからこれらのチャンネルにメッセージを送信します。
4. **Nickname** フィールドに、指定されたチャンネルにメッセージを送信する IRC ニックネームを指定します。
 5. 指定したニックネームにパスワードがある場合は、**NickServ password** フィールドにそのパスワードを指定します。
 6. **Next** をクリックしてアクションの入力タイプを指定します。
 7. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
 8. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 9. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 10. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 11. **Next** をクリックして、コネクションをフローに追加します。

結果

フロービジュアライゼーションで追加した場所にコネクションが表示されます。

第13章 JIRA への接続

インテグレーションでは、Jira サーバーに接続して課題を取得、作成、および更新できます。Jira はソフトウェア開発プロジェクトを計画、追跡、および管理するためのツールです。各 Jira 環境では、ワークフローをカスタマイズでき、これはインテグレーションで Jira に接続するための詳細に影響します。Jira 管理者に相談して詳細を明確にしてください。

Jira サーバーへ接続するインテグレーションを作成するための情報および手順は、以下を参照してください。

- [「OAuth を使用する Jira サーバーで Fuse Online を登録」](#)
- [「Jira コネクションの作成」](#)
- [「Jira の課題またはコメントを取得してインテグレーションの実行をトリガー」](#)
- [「課題を Jira サーバーに追加」](#)
- [「Jira の課題へのコメントの追加」](#)
- [「Jira の課題へのファイルの添付」](#)
- [「Jira 課題のウォッチャーの追加および削除」](#)
- [「Jira サーバーでの課題の更新」](#)
- [「Jira の課題の新しいステータスへのトランジション」](#)

13.1. OAUTH を使用する JIRA サーバーで FUSE ONLINE を登録

インテグレーションは、アクセスの認証に OAuth プロトコルを使用する Jira サーバーに接続できます。これを有効にするには、Fuse Online 環境を Jira クライアントアプリケーションとして登録します。Jira クライアントアプリケーションは、Jira コンシューマーとも呼ばれます。つまり、Fuse Online 環境は Jira コンシューマーになります。



注記

Jira サーバーが Basic 認証のみを使用する場合、登録は必要ありません。

ここで説明する手順にしたがって、Jira への Fuse Online コネクションを作成するために必要な Jira OAuth クレデンシャルを取得します。サンプルサーバーで手順を説明するチュートリアルなど、Jira コンシューマーの登録に関する詳細は、Jira ドキュメントの [OAuth for REST APIs](#) を参照してください。

Jira サーバーで Fuse Online を登録した後、Fuse Online を使用してその Jira サーバーへのコネクションを作成します。任意の数のインテグレーションが同じ Jira コネクションを使用できます。しかし、別の Jira コネクションを作成する場合は、ここで説明する登録手順を再度行い、別の OAuth クレデンシャルのセットを取得する必要があります。

主なステップの概要

登録手順の主なステップは次のとおりです。

- OAuth クレデンシャルの取得を可能にする Atlassian コマンドラインアプリケーションを取得します。
- ダウンロードしたアプリケーションを実行して **config.properties** ファイルを作成します。

- Fuse Online が Jira サーバーへのアクセスに使用する公開鍵と秘密鍵のペアを作成します。
- **config.properties** ファイルを編集し、値の一部を更新します。
- Jira サーバーを設定し、Fuse Online 環境がコンシューマーアプリケーションとして認識されるようにします。
- Fuse Online が Jira サーバーでデータにアクセスするために必要な Jira アクセストークンを取得します。

これらの各ステップには複数のサブステップが含まれ、以下の手順で詳細を提供します。

前提条件

- Jira サーバー管理者権限が必要です。
- コマンドラインインターフェイスの使い方を知っている必要があります。

手順

1. OAuth クレデンシャルの取得を可能にする Atlassian コマンドラインアプリケーションを取得します。
 - a. [atlassian-oauth-examples](#) ページに移動します。
 - b. 左側のパネルの下にある **Downloads** をクリックします。
 - c. **Downloads** ページの **Download repository** をクリックします。
 - d. 任意のディレクトリーでファイルを展開します。
2. ダウンロードしたアプリケーションを実行して OAuth クレデンシャルの **config.properties** ファイルを作成します。
 - a. ターミナルで、展開した内容が含まれるディレクトリーに移動します。
 - b. **java** ディレクトリーに移動し、クライアントをビルドします。

```
cd atlassianlabs-atlassian-oauth-examples-dd0264cad043/java
mvn clean compile assembly:single
```

- c. **target** ディレクトリーに移動し、**config.properties** ファイルを生成します。

```
cd target
java -jar OAuthTutorialClient-1.0.jar requestToken
```

コネクション例外は無視しても構いません。**target** ディレクトリーに、一時キーの一部が含まれる **config.properties** ファイルが作成されます。

3. Fuse Online が Jira サーバーをアクセスするのに必要な公開鍵と秘密鍵のペアを生成します。これらのコマンドを実行する場所は自由ですが、便宜上の理由で最初に特定のディレクトリーを指示されます。
 - a. ダウンロードしたアプリケーションが含まれるディレクトリーに移動します。
 - b. 秘密鍵を生成します。

-

```
openssl genrsa -out jira_privatekey.pem 1024
```

- c. 公開鍵を取得します。

```
openssl req -newkey rsa:1024 -x509 -key jira_privatekey.pem -out jira_publickey.cer -
days 365
```

このコマンドでは、Fuse Online 環境に関する情報を入力するよう要求されます。プロンプトの応答を省略するには、このコマンドに **-subj** オプションを追加します。たとえば、上記のコマンドラインの最後に、以下を追加します。

```
-subj "/C=US/ST=New York/L=New York/O=MyCompany
Ltd./OU=IT/CN=mysubdomain.mydomain.com"
```

- d. 鍵をエクスポートします。

```
openssl pkcs8 -topk8 -nocrypt -in jira_privatekey.pem -out jira_privatekey.pcks8
openssl x509 -pubkey -noout -in jira_publickey.cer > jira_publickey.pem
```

4つのファイルが存在します。

```
jira_privatekey.pcks8
jira_privatekey.pem
jira_publickey.cer
jira_publickey.pem
```

- e. ヘッダー、改行、およびフッターなしで秘密鍵を取得します。

```
cat jira_privatekey.pcks8 | grep -v 'PRIVATE'|awk '{print}' ORS=""
```

- f. 秘密鍵をクリップボードにコピーします。

4. **config.properties** ファイルを編集します。

- config.properties** ファイルが含まれる **java/target** ディレクトリーに移動し、ファイルを開いて編集します。
- config.properties** ファイルの秘密鍵を、クリップボードにコピーした秘密鍵に置き換えます。
- jira_home** の設定を Jira サーバーの URL に変更します。Jira サーバーが **https** を使用する場合は必ず **https** を指定してください。また、コロンをエスケープ処理する必要があります。以下に例を示します。


```
jira_home=https\://issues.mycompany.org
```

- d. **consumer_key** 設定を **Fuse Online App** に変更します。以下に例を示します。

```
consumer_key=Fuse Online App
```

- e. **config.properties** ファイルを保存して、閉じます。

- Jira サーバーを設定し、Fuse Online 環境がコンシューマーアプリケーションとして認識されるようにします。管理ユーザーとして Jira サーバーにログインする必要があります。

- a. **Settings** () > **Products** > **Application links** と選択します。
 - b. **Create new link** 入力フィールドに URL を入力します。たとえば、 <https://example.com> を入力します。
これにより、受信された応答がないことを示すダイアログが表示されます。1方向通信のみが必要であるため、問題はありません。
 - c. **Continue** をクリックすると **Link applications** ダイアログが表示されます。このダイアログで以下を行います。
 - i. **Consumer Key** フィールドに **Fuse Online App** を入力します。
 - ii. その下で **Create incoming link** を選択します。
 - iii. 他のフィールドにはそれぞれ任意の値を入力します。すべてのフィールドに値が必要です。応答は必要ないため、何を入力しても構いません。
 - iv. **Continue** をクリックすると、別のダイアログが表示されます。
 - v. **Consumer Key** フィールドに **Fuse Online App** を入力します。
 - vi. **Consumer Name** フィールドに **Fuse Online App** を入力します。
 - vii. **Public Key** フィールドに **jira_publickey.pem** ファイルの公開鍵を貼り付けます。ヘッダーとフッターのない鍵のみを貼り付けるようにしてください。コピーするために鍵を表示するには、**jira_publickey.pem** ファイルが含まれるディレクトリーで **cat jira_publickey.pem** を実行します。
 - viii. **Continue** をクリックします。
6. Fuse Online が Jira サーバーでデータにアクセスするために必要な Jira アクセストークンを取得します。この手順の最初にリンクした Jira OAuth ドキュメントでは、このステップは The OAuth dance と呼ばれます。これは、Fuse Online、Jira サーバー、および Jira 管理者の間でやり取りがあるためです。
- a. この手順の最初のステップでダウンロードしたコマンドラインアプリケーションが含まれるディレクトリーで、**java/target** ディレクトリーに移動します。
 - b. Jira サーバーからリクエストトークンを取得します。

```
java -jar OAuthTutorialClient-1.0.jar requestToken
```

以下のような出力が表示されます。

```
Token:      ec3dj4byySM5ek3XW7gl7f4oc99obAlo
Token Secret: OhONj0eF7zhXAMKZLbD2Rd3x7Dmxjy0d
Retrieved request token. go to
https://jira101.atlassian.net/plugins/servlet/oauth/authorize?
oauth_token=ec3dj4byySM5ek3XW7gl7f4oc99obAlo to authorize it
```
 - c. ブラウザーの別のタブで、この出力にある URL に移動します。Jira のウェルカムページが表示され、Fuse Online App へのアクセスを許可するよう要求されます。
 - d. **Allow** をクリックすると、確認コードのある **Access Approved** ページが表示されます。
 - e. 確認コードをクリックボードにコピーします。簡単に使える場所に貼り付けます。

- f. 以下のようなコマンドを実行し、アクセストークンを取得します。以下のコマンドラインの例では、最後の値が確認コードになります。例の確認コードを前のステップでコピーした確認コードに置き換えてからコマンドを実行します。以下に例を示します。

```
java -jar OAuthTutorialClient-1.0.jar accessToken qTJkPi
```

このコマンドは、Fuse Online が Jira サーバーにアクセスするのに必要なアクセストークンを出力します。

結果

Jira コマンドラインアプリケーションは **config.properties** ファイルを更新し、Fuse Online の Jira コネクションの設定に必要なすべての値が含まれるようにします。

次のステップ

Jira コネクションを作成します。

13.2. JIRA コネクションの作成

インテグレーションで、Jira の課題を取得、作成、または更新するには、インテグレーションに追加する Jira コネクションを作成します。同じコネクションを任意の数のインテグレーションに追加できます。

接続する Jira サーバーがアクセスの認証に OAuth プロトコルを使用する場合、接続する Jira サーバーのクライアントとして Fuse Online 環境を登録する必要があります。[OAuth を使用する Jira サーバーで Fuse Online を登録](#) を参照してください。

前提条件

- Basic 認証を使用する Jira サーバーでは、Jira のユーザー名とパスワードがあります。
- OAuth を使用する Jira サーバーでは、Jira アクセストークン、コンシューマーキー、秘密鍵、および確認コードの値が含まれる **config.properties** ファイルにアクセスできます。

手順

1. Fuse Online のナビゲーションパネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックし、利用可能なコネクターを表示します。
3. **Jira** コネクターをクリックします。
4. **Jira URL** フィールドに、インテグレーションが接続する Jira サーバーの URL を入力します。たとえば <https://issues.mycompany.org> を入力します。
Jira サーバーへのコネクションが OAuth を使用する場合、Fuse Online 環境を Jira コンシューマーアプリケーションとして登録する手順で **config.properties** ファイルが作成されます。ここで指定する URL は **config.preperities** ファイルにある URL と同じである必要があります。たとえば、**config.properties** ファイルに以下のような行があります。

```
jira_home=https://issues.mycompany.org
```

5. この Jira サーバーが Basic 認証を使用する場合は、以下を行います。

- a. **User name** フィールドに Jira ユーザー名を入力します。接続はこのアカウントを使用して、指定の Jira サーバーで動作します。
- b. **Password** フィールドに、指定したユーザーのパスワードを入力します。

Jira サーバーが OAuth 認証を使用しない場合は、次のステップをスキップします。

6. Jira サーバーが OAuth 認証を使用する場合は、以下を行います。
 - a. OAuth アクセストークン、キー、および確認コードが含まれる **config.properties** ファイルを開きます。これは、登録の手順で作成されたファイルです。そのファイルから値をコピーし、接続設定フィールドに貼り付けます。
 - b. **access_token** の値を **Access Token** フィールドにコピーします。
 - c. **consumer_key** の値を **Consumer Key** フィールドにコピーします。
 - d. **private_key** の値を **Private Key** フィールドにコピーします。
 - e. **secret** の値を **Verification Code** フィールドにコピーします。
7. **Validate** をクリックします。Fuse Online はこの接続を検証できるかを示すメッセージを表示します。検証に失敗した場合は、入力した値が正しいことを確認し、再度検証を行ってください。
8. 検証に成功したら **Next** をクリックします。
9. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**Jira Connect 1** を入力します。
10. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。たとえば、**Sample Jira connection that uses the jkim account.** を入力します。
11. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **Jira Connect 1** が表示されます。

13.3. JIRA の課題またはコメントを取得してインテグレーションの実行をトリガー

Jira サーバーから課題またはコメントを取得したときにインテグレーションの実行をトリガーするには、Jira 接続をシンプルなインテグレーションの最初の接続として追加します。

前提条件

- Jira 接続が作成済みである必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する Jira 接続をクリックします。

4. **Choose an action** ページで、以下のアクションの1つを選択します。

- **Retrieve New Comments** は指定したクエリーを使用して Jira サーバーを検索します。前回クエリーが呼び出された後、クエリーが返す各課題にコメントが追加されたかどうかはコネクションによって確認されます。新しいコメントがコネクションによって返されません。
- **Retrieve New Issues** は、指定したクエリーを使用して Jira サーバーを検索します。クエリーが返す各課題の作成日がコネクションによって確認されます。前回のクエリーが呼び出された後に課題が作成された場合、コネクションはその課題をインテグレーションに戻します。コネクションがこのクエリーを初めて実行した場合は、見つかったすべての課題を返します。

5. **Delay** フィールドにデフォルトの 5000 ミリ秒 (5 秒) を使用するか、クエリー呼び出しの間の待ち時間をミリ秒単位で指定します。

6. **Jira Query Language** フィールドに、Jira クエリーを入力します。以下に例を示します。

```
project = GATE AND assignee = jkim AND (status = Open OR status = New OR status = Reopened)
```

このクエリーは、**jkim** に割り当てられた未解決の **Gateway Online** プロジェクトの課題を検索します。各 Jira プロジェクトには短い名前と長い名前があります。この例では、**Gateway Online** プロジェクトの短い名前は **GATE** になります。クエリーではプロジェクトの短い名前を指定します。

クエリーが具体的であればあるほど、Jira サーバーでの検索時間が短縮されます。クエリーの結果がない場合、コネクションはインテグレーションの実行をトリガーしません。

7. **Next** をクリックします。

結果

コネクションはシンプルなインテグレーションを開始し、Fuse Online はインテグレーションの最後のコネクションを選択するよう要求します。

13.4. 課題を JIRA サーバーに追加

Jira サーバーに接続してフローの途中で課題を追加したり、シンプルなインテグレーションを終了するために Jira サーバーに接続して課題を追加することができます。

前提条件

- Jira コネクションが作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online でシンプルなインテグレーションの最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、Jira コネクションの追加先のプラス記号をクリックします。シンプルなインテグレーションの最後のコネクションを追加する場合は、このステップをスキップします。
2. インテグレーションが Jira サーバーへの接続に使用する Jira コネクションをクリックします。

3. **Choose an action** ページで **Add Issue** を選択します。
4. アクションを設定します。必須のフィールドは **Issue type ID or name** と **Project key** です。任意のフィールドには現時点で値を入力できますが、このコネクションをインテグレーションに追加した後に、このコネクションの前にデータマッピングステップを追加することもできます。データマッピングステップでは、前のステップのフィールドをこのコネクションの任意のフィールドにマップします。
 - a. 任意設定: **Assignee** フィールドにこの課題を割り当てる担当者の Jira ユーザー名を入力します。
 - b. 任意設定: **Components** フィールドに、1つ以上のコンポーネントの名前を入力します。複数の名前はコンマで区切ります。指定できるコンポーネントは Jira 管理者が定義します。
 - c. 任意設定: **Priority ID or name** フィールドに、新しい課題の優先度を入力します。入力できる優先度の値は Jira 管理者が定義します。
 - d. 任意設定: **Summary** フィールドに、この課題の概要を入力します。
 - e. **Issue type ID or name** フィールドに、課題のタイプを表す値を入力します。たとえば、**Story**、**Epic**、**Chore**、**Bug** などが通常の値になります。
Jira では、管理者権限を持つユーザーが **Project settings > Issue types** と選択すると、有効な課題タイプをリストで表示できます。この権限を持たないユーザーは、課題のタイプフィールドをクリックすると、許可される値のリストを表示できます。
 - f. 任意設定: **Watchers** フィールドに、Jira ユーザー名のコンマ区切りリストを入力します。課題が更新されると、これらのユーザーに通知が送信されます。
 - g. **Project Key** フィールドに、Jira 管理者が定義した Jira プロジェクトの短い名前を入力します。コネクションは、このプロジェクトで新しい課題を作成します。たとえば、プロジェクトキーは **GATE** のようになります。
 - h. 任意設定: **Description** フィールドに、この課題に関する便利な情報を入力します。
5. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

13.5. JIRA の課題へのコメントの追加

Jira サーバーに接続してフローの途中でコメントを追加したり、シンプルなインテグレーションを終了するために Jira サーバーに接続してコメントを追加することができます。

前提条件

- Jira コネクションが作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online でシンプルなインテグレーションの最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、Jira コネクションの追加先のプラス記号をクリックします。シンプルなインテグレーションの最後のコネクションを追加する場合は、このステップをスキップします。
2. インテグレーションが Jira サーバーへの接続に使用する Jira コネクションをクリックします。
3. **Choose an action** ページで **Add Comment** を選択します。
4. **Comment** フィールドに、課題に追加するコメントを入力するか、このコネクションの前にデータマッピングステップを追加するよう計画します。データマッピングステップでは、前のインテグレーションステップのコメントをこのコネクションの **comment** フィールドにマップします。
5. **Issue Key** フィールドに、課題の Jira 識別子を入力します。たとえば、課題キーは **GATE-1234** や **Ent-239** のようになります。
6. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

次のステップ

前のインテグレーションステップからこのコネクションにコメントをマップするには、コメントの内容を提供するコネクションを追加し、Jira コネクションの前にデータマッピングステップを追加します。たとえば、データベースコネクションで顧客の記録を取得する場合に、データマッピングステップで顧客の記録のコレクションを Jira コネクションの **comment** フィールドにマップします。

13.6. JIRA の課題へのファイルの添付

Jira サーバーに接続してフローの途中で1つ以上のファイルを課題に添付したり、Jira サーバーに接続し1つ以上のファイルを課題に添付してシンプルなインテグレーションを終了することができます。インテグレーションでは、添付するファイルを取得する事前のステップが必要です。

前提条件

- Jira コネクションが作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online でシンプルなインテグレーションの最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、Jira コネクションの追加先のプラス記号をクリックします。シンプルなインテグレーションの最後のコネクションを追加する場合は、このステップをスキップします。
2. インテグレーションが Jira サーバーへの接続に使用する Jira コネクションをクリックします。
3. **Choose an action** ページで **Attach File** を選択します。
4. **Issue Key** フィールドに、ファイルに添付する課題の Jira 識別子を入力します。たとえば、課題キーは **GATE-1234** や **Ent-239** のようになります。
5. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。

次のステップ

Jira の課題を添付するファイルを取得する初期の接続がインテグレーションにない場合、その接続をインテグレーションに追加します。その接続は、ファイルを添付する Jira 接続の前に追加してください。たとえば、FTP 接続を追加し、FTP サーバーから1つ以上のファイルをダウンロードする場合に、インテグレーションが実行されると、後続の Jira 接続は、Attach File アクションの設定時に指定した Jira の課題にこれらのファイルを添付します。

13.7. JIRA 課題のウォッチャーの追加および削除

Jira サーバーに接続してフローの途中で課題のウォッチャーを追加または削除したり、シンプルなインテグレーションを終了するために Jira サーバーに接続して課題のウォッチャーを追加または削除することができます。

前提条件

- Jira 接続が作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online でシンプルなインテグレーションの最後の接続を選択するよう要求されます。

手順

1. **Add to Integration** ページで、Jira 接続の追加先のプラス記号をクリックします。シンプルなインテグレーションの最後の接続を追加する場合は、このステップをスキップします。
2. インテグレーションが使用する Jira 接続をクリックします。
3. **Choose an action** ページで **Add Issue** を選択します。
4. **Issue Key** フィールドに、ウォッチャーと追加または削除する課題の Jira 識別子を入力します。たとえば、課題キーは **GATE-1234** や **Ent-239** のようになります。
5. 任意設定: **Add Watchers** フィールドに、Jira ユーザー名のコンマ区切りリストを入力します。課題が更新されると、これらのユーザーに通知が送信されます。
6. 任意設定: **Remove Watchers** フィールドに、課題のウォッチャーである Jira ユーザー名のコンマ区切りリストを入力します。接続はこれらのユーザーをウォッチャーのリストから削除します。
7. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。

13.8. JIRA サーバーでの課題の更新

フローの途中で Jira サーバーに接続して課題を更新したり、シンプルなインテグレーションを終了するために Jira サーバーに接続して課題を更新することができます。接続は、1つまたは複数の課

題フィールドを更新できます。更新した課題にフィールドの値がある場合、その値は接続に指定した値や、接続フィールドにマップした値に置き換えられます。たとえば、課題に説明がすでにあり、**Update Issue** の設定時に **Description** フィールドにテキストを指定すると、接続は、既存の説明を、新しい説明に置き換えます。

前提条件

- Jira 接続が作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online でシンプルなインテグレーションの最後の接続を選択するよう要求されます。

手順

1. **Add to Integration** ページで、Jira 接続の追加先のプラス記号をクリックします。シンプルなインテグレーションの最後の接続を追加する場合は、このステップをスキップします。
2. インテグレーションが Jira サーバーへの接続に使用する Jira 接続をクリックします。
3. **Choose an action** ページで **Update Issue** を選択します。
4. アクションを設定します。必須のフィールドは **Issue Key** です。任意のフィールドには現時点で値を入力できますが、この接続をインテグレーションに追加した後に、この接続の前にデータマッパーステップを追加することもできます。データマッパーステップでは、前のステップのフィールドをこの接続の任意のフィールドにマップします。
 - a. 任意設定: **Assignee** フィールドにこの課題を割り当てる担当者の Jira ユーザー名を入力します。
 - b. 任意設定: **Components** フィールドに、1つ以上のコンポーネントの名前を入力します。複数の名前はコンマで区切ります。指定できるコンポーネントは Jira 管理者が定義します。
 - c. **Issue Key** フィールドに、接続が更新する課題の Jira 識別子を入力します。たとえば、課題キーは **GATE-1234** や **Ent-239** のようになります。
 - d. 任意設定: **Priority ID or name** フィールドに、新しい課題の優先度を入力します。入力できる優先度の値は Jira 管理者が定義します。
 - e. 任意設定: **Summary** フィールドに、この課題の概要を入力します。
 - f. **Issue type ID or name** フィールドに、課題のタイプを表す値を入力します。たとえば、**Story**、**Epic**、**Chore**、**Bug** などが通常の値になります。
 - g. 任意設定: **Description** フィールドに、この課題に関する便利な情報を入力します。
5. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。

13.9. JIRA の課題の新しいステータスへのトランジション

フローの途中で Jira サーバーに接続して課題を新しいステータスや新しいワークフローステージにトランジションしたり、シンプルなインテグレーションを終了するために Jira サーバーに接続して課題を新

しいステータスや新しいワークフローステージにトランジションすることができます。Jira 管理者は、課題のトランジションの意味や、トランジション ID の値を定義します。

前提条件

- Jira コネクションが作成済みである必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online でシンプルなインテグレーションの最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、Jira コネクションの追加先のプラス記号をクリックします。シンプルなインテグレーションの最後のコネクションを追加する場合は、このステップをスキップします。
2. インテグレーションが Jira サーバーへの接続に使用する Jira コネクションをクリックします。
3. **Choose an action** ページで **Transition Issue** を選択します。
Transition Issue アクションの設定には、値を指定する必要はありません。値を指定する代わりに、前のインテグレーションステップからこのコネクションのフィールドに値をマップできます。
4. 任意設定: **Comment** フィールドに、課題に追加するコメントを入力します。
5. 任意設定: **Issue Key** フィールドに、トランジションを行う課題の Jira 識別子を入力します。たとえば、課題キーは **GATE-1234** や **Ent-239** のようになります。
6. 任意設定: **ID of the transition** フィールドに、Jira 管理者が特定のトランジションに関連付けた整数を入力します (Jira が新規から作業中に移行したり、作業中からテスト中に移行した場合など)。
Jira 管理者は、プロジェクトのワークフロー表示でトランジション ID を確認できます。管理権限がないユーザーは、Jira 管理者から許可されたトランジション ID を取得します。
7. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

次のステップ

各 **Transition Issue** フィールドに値を指定しなかった場合、インテグレーションに各フィールドの値を取得するステップがあるようにしてください。インテグレーションに必要なコネクションがすべて追加された後、課題のトランジションを行う Jira コネクションの前にデータマッピングステップを追加します。データマッピングステップで、前のステップから課題のトランジションを行う Jira コネクションにフィールドをマップします。

第14章 KAFKA への接続

Apache Kafka はデータの取得およびパブリッシュに使用できる分散ストリーミングプラットフォームです。インテグレーションでは、指定した Kafka トピックからデータをサブスクライブでき、または指定した Kafka トピックにデータをパブリッシュすることもできます。これには、Kafka へのコネクションを作成し、そのコネクションをインテグレーションフローに追加します。詳細は以下を参照してください。

- [「Kafka ブローカーへのコネクションの作成」](#)
- [「Kafka ブローカーからデータを取得してインテグレーションの実行をトリガー」](#)
- [「Kafka ブローカーへのデータの公開」](#)

14.1. KAFKA ブローカーへのコネクションの作成

インテグレーションで、Kafka トピックからデータをサブスクライブしたり、Kafka トピックにデータを公開するには、Kafka へのコネクションを作成し、そのコネクションをインテグレーションに追加します。



重要

本リリースでは、Kafka へのコネクションは SSL をサポートしません。これは、今後のリリースで変更される予定です。

前提条件

接続する Kafka ブローカーの URI を知っている必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックしてコネクタを表示します。
3. **Kafka** コネクタをクリックします。
4. **Kafka bootstraps URI** フィールドに、Kafka ブローカー URI のコンマ区切りリストを入力します。各 URI は `host:port` 形式で指定する必要があります。
5. **Validate** をクリックします。Fuse Online は即座にコネクションを検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、入力パラメーターを訂正し、再度検証を行います。
6. 検証に成功した場合は **Next** をクリックします。
7. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Kafka West** を入力します。
8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。
9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Kafka West** が表示されます。

14.2. KAFKA ブローカーからデータを取得してインテグレーションの実行をトリガー

Kafka ブローカーからデータを受信したときにインテグレーションの実行をトリガーするには、Kafka コネクションを最初のコネクションとして追加します。インテグレーションの実行中、Kafka コネクションは指定した Kafka トピックでデータを継続的に監視します。コネクションが新しいデータを見つけると、そのデータをインテグレーションの次のステップに渡します。

前提条件

Kafka ブローカーへのコネクションが作成されている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する Kafka コネクションをクリックします。
4. **Choose an action** ページで、**Subscribe** アクションを選択し、指定したトピックからデータを受信します。
5. **Topic Name** フィールドで、下向きの矢印をクリックしてトピックのリストを表示し、サブスクライブするトピックをクリックします。
6. **Next** をクリックして、アクションの出力タイプを指定します。
7. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
8. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
9. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときにこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。

10. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
11. **Next** をクリックします。

結果

インテグレーションに最初のコネクションが追加され、Fuse Online は最後のコネクションを選択するよう要求します。

14.3. KAFKA ブローカーへのデータの公開

インテグレーションでは、フローの途中で Kafka ブローカーにデータをパブリッシュしたり、シンプルなインテグレーションを終了するためにデータをパブリッシュしたりすることができます。これには、Kafka コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Kafka ブローカーへのコネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、Kafka コネクションの追加先のプラス記号をクリックします。シンプルなインテグレーションの最後のコネクションを追加する場合は、このステップをスキップします。
2. メッセージの公開に使用する Kafka コネクションをクリックします。
3. **Choose an action** ページで **Publish** を選択します。
4. **Topic Name** フィールドで、下向きの矢印をクリックしてトピックのリストを表示し、公開するトピックをクリックします。
5. **Next** をクリックしてアクションの入力タイプを指定します。
6. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。

7. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
8. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
9. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
10. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

第15章 KNATIVE リソースへの接続

以下のいずれかを行うシンプルなインテグレーションを作成できます。

- Knative チャンネルに接続し、メッセージを取得または送信。
- インテグレーションを Knative サービスとして公開。
- Knative サービスを呼び出し。



重要

Knative リソースへの接続はテクノロジープレビューの機能です。テクノロジープレビュー機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、Red Hat は本番環境での使用は推奨しません。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能により、近日発表予定の製品機能をリリースに先駆けてご提供でき、お客様は開発プロセス時に機能をテストして、フィードバックをお寄せいただくことができます。Red Hat のテクノロジープレビュー機能のサポート範囲に関する詳細は、<https://access.redhat.com/ja/support/offerings/techpreview> を参照してください。

変換されたメッセージの形式に規定はありませんが、メッセージは通常 JSON 形式になります。

たとえば、シンプルなインテグレーションは Salesforce を定期的にポーリングして新しいリードを探すことがあります。Salesforce コネクションが新しいリードを見つけた場合、詳細が含まれるメッセージを Knative チャンネルに送信することができます。複数のシンプルなインテグレーションは同じ Knative チャンネルからメッセージを取得することができます。各インテグレーションはリードデータに異なるアクションを実行することができます。

詳細は以下のトピックを参照してください。

- 「[Knative リソースへのコネクションを有効化](#)」
- 「[Knative チャンネルにサブスクライブしてメッセージを取得](#)」
- 「[Knative チャンネルへのメッセージの送信](#)」
- 「[シンプルなインテグレーションを Knative サービスとして公開](#)」
- 「[シンプルなインテグレーションからの Knative サービスの呼び出し](#)」

15.1. KNATIVE リソースへのコネクションを有効化

Knative コネクションを有効にしてからシンプルなインテグレーションに追加する必要があります。

前提条件

- Fuse Online が実行している OCP クラスタに **oc** クライアントツールがインストールされ、接続されている必要があります。
- お客様が管理する環境で Fuse Online がオンサイトの OpenShift Container Platform で稼働している必要があります。
- Fuse Online のランタイムが Camel K である必要があります。

- Fuse Online がインストールされているクラスターに、インテグレーションが接続する各 Knative リソースもインストールされている必要があります。
- Fuse Online をインストールできる権限が必要です。

手順

1. 以下のコマンドを実行して、Fuse Online サーバーを 0 にスケールします。

```
oc scale dc/syndesis-operator --replicas 0
```

2. **syndesis-server** ConfigMap ファイルを編集し、**features.knative.enabled** を **true** に設定します。oc CLI を使用するには、**oc edit cm syndesis-server** を実行します。
3. Camel K **IntegrationPlatform** リソースを編集して、**spec.profile** を **Knative** に設定します。oc CLI を使用するには、**oc edit integrationplatform** を実行します。
4. 以下のコマンドを実行して、Fuse Online サーバー(**syndesis-server**)リソースを再起動します。

```
oc delete pod -l syndesis.io/component=syndesis-server
```

結果

Fuse Online コンソールの **Connections** ページで Knative コネクションが利用可能になります。同じ Knative コネクションを任意の数のシンプルなインテグレーションに追加できます。コネクションアクションの設定は、コネクションがアクセスする Knative リソースを指定します。

15.2. KNATIVE チャネルにサブスクライブしてメッセージを取得

Knative チャネルからメッセージを受信したときにインテグレーションの実行をトリガーするには、Knative コネクションをシンプルなインテグレーションの最初のコネクションとして追加します。このように開始するインテグレーションは、指定期間内にチャネルからメッセージを受信しないと自動的に 0 Pod にスケールダウンされます。デフォルトの期間は1分です。

この自動スケールリングはリソースを節約します。そのため、AMQ ブローカーへのコネクションなどの、他のコネクションで始まるシンプルなインテグレーションよりも優れています。

前提条件

- Knative リソースへのコネクションが有効になっている必要があります。
- メッセージを取得する Knative リソースが Fuse Online と同じクラスターにある必要があります。
- コネクションがメッセージを取得する Knative チャネルの名前を知っている必要があります。

手順

1. Fuse Online の左側のパネルで **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで Knative コネクションをクリックします。

4. **Receive from Channel** アクションを選択します。
5. **Channel name** フィールドをクリックし、使用できる Knative チャネルの一覧を表示します。
6. メッセージを取得するリソースの Knative チャネル名を選択します。
7. **Next** をクリックし、**Receive from Channel** アクションの出力タイプを指定します。これは、インテグレーションでコネクションが次のステップに渡すタイプです。
 - a. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
 - b. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 - c. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 - d. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 - e. **Next** をクリックします。

結果

Fuse Online は、シンプルなインテグレーションの最後のコネクションを追加するよう要求します。

実行中、選択した Knative チャネルが新しいメッセージを受信するたびに、Knative コネクションはインテグレーションの次のステップにメッセージを渡します。

次のステップ

必要なすべてのコネクションをインテグレーションに追加します。その後、インテグレーションを開始する Knative コネクションの後にデータマッパーステップを追加します。ソース **message** フィールドを、次のコネクションのターゲットフィールドにマップマップします。

15.3. KNATIVE チャネルへのメッセージの送信

シンプルなインテグレーションでは、Knative コネクションはフローの途中で Knative チャネルにメッセージを送信したり、シンプルなインテグレーションを終了するためにメッセージを送信することができます。これには、Knative コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Knative コネクションが有効になっている必要があります。
- メッセージの送信先となる Knative リソースが Fuse Online と同じクラスターにインストールされている必要があります。
- コネクションがメッセージを送信する Knative チャネルの名前を知っている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. **Choose a connection** ページで Knative コネクションをクリックします。
3. **Send to Channel** アクションを選択します。
4. **Channel name** フィールドをクリックし、Fuse Online を実行しているクラスターで使用できる Knative チャネルの一覧を表示します。
5. インテグレーションがメッセージを送信するチャネルを選択します。
6. **Next** をクリックし、**Send to Channel** アクションの入力タイプを指定します。これは、インテグレーションが Knative チャネルに送信するタイプです。
 - a. データタイプの認識が必要がなかったり、前のインテグレーションステップからの出力が想定されたタイプであった場合、**Select Type** フィールドで **Type specification not required** を指定し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
 - b. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。

- c. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
- d. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
- e. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで、追加した場所にコネクションが表示されます。実行中、このコネクションは選択した Knative チャンネルにメッセージを送信します。

次のステップ

必要なすべてのコネクションをインテグレーションに追加します。次に、Knative チャンネルにメッセージを送信する Knative コネクションの直前にデータマッパーステップを追加します。ソースフィールドを Knative コネクションターゲット **message** フィールドにマップします。

15.4. シンプルなインテグレーションを KNATIVE サービスとして公開

シンプルなインテグレーションでは、インテグレーションを Knative サービスとして公開する Knative コネクションを最初のコネクションとすることができます。このように開始するインテグレーションは、指定の期間サービスに呼び出しがないと自動的に 0 Pod にスケールダウンします。デフォルトの期間は 1 分です。

この自動スケーリングはリソースを節約します。そのため、AMQ ブローカーへのコネクションなどの、他のコネクションで始まるシンプルなインテグレーションよりも優れています。

前提条件

- Knative リソースへのコネクションが有効になっている必要があります。
- Knative が Fuse Online が稼働しているクラスターにインストールされている必要があります。

手順

1. Fuse Online の左側のパネルで **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで Knative コネクションをクリックします。
4. **Expose as Service** アクションを選択します。

結果

Fuse Online は、シンプルなインテグレーションの最後のコネクションを追加するよう要求します。

Fuse Online がインテグレーションをパブリッシュするとき、Camel K ランタイムは Knative **servicing.knative.dev** サービスを作成します。このサービスは以下のような特徴があります。

- Kubernetes デプロイメントオブジェクトの代替です。
- 一定期間アクティビティーがないと、自動的に 0 にスケーリングできます。
- Kubernetes サービスと混同しないようにしてください。

パブリッシュすると、ルートパス (/) でインテグレーションを呼び出すことが可能になります。クライアントは、インテグレーションを HTTP エンドポイントとして呼び出し、必要に応じてデータを渡すことで、インテグレーションの実行を呼び出します。通常、渡されたデータは JSON 形式になります。

本リリースでは、インテグレーションを Knative サービスとして公開する主な目的は、他の Fuse Online インテグレーションがその Knative サービスを呼び出すことであると想定されます。この場合、Knative サービスを呼び出すインテグレーションがサービス名を選択し、パスを指定する必要はありません。

15.5. シンプルなインテグレーションからの KNATIVE サービスの呼び出し

シンプルなインテグレーションでは、Knative コネクションはフローの途中で Knative サービスを呼び出したり、シンプルなインテグレーションを終了するために Knative サービスを呼び出すことができます。これには、Knative コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Knative コネクションが有効になっている必要があります。
- コネクションが呼び出す Knative サービスの名前を知っておく必要があります。このサービスは Fuse Online と同じクラスターで実行されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. **Choose a connection** ページで Knative コネクションをクリックします。
3. **Call Service** アクションを選択します。
4. **Service name** フィールドをクリックし、Fuse Online が実行しえているクラスターで使用できる Knative サービスの一覧を表示します。
5. インテグレーションが呼び出すサービスを選択します。
呼び出す Knative サービスがインテグレーションとして公開されている場合、サービスの名前はインテグレーションと同じ名前または似た名前になります。
6. **Next** をクリックし、**Call Service** アクションの入力タイプを指定します。これは、インテグレーションが Knative サービスに送信するタイプです。
 - a. データタイプの認識が必要がなかったり、前のインテグレーションステップからの出力が想定されたタイプであった場合、**Select Type** フィールドで **Type specification not required** を指定し、**Next** をクリックします。残りの手順を行う必要はありません。

これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。

- **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
- b. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
- c. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
- d. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
- e. **Next** をクリックします。

7. **Call Service** アクションの出力タイプを指定します。

これは、Knative コネクションが途中のコネクションである場合に Knative サービスがインテグレーションに返すタイプです。

前の手順を繰り返して出力タイプを指定します。または、Knative サービスの応答が空である場合は **Type specification not required** を指定します。

8. **Next** をクリックしてコネクションを追加します。

結果

インテグレーションビジュアライゼーションで、追加した場所にコネクションが表示されます。

実行中、このコネクションは指定の Knative サービスを呼び出します。呼び出しには、前のインテグレーションステップ (この Knative コネクションの前のステップ) からの出力が含まれます。Knative コネクションが途中のコネクションである場合は、コネクションはサービスの出力をインテグレーションの次のステップに渡します。

次のステップ

必要な他のコネクションをインテグレーションに追加します。入力タイプや出力タイプを指定した場合、Knative コネクションの前や後にデータマッパーステップを追加し、必要に応じてフィールドをマップします。

第16章 APACHE KUDU への接続

Apache Kudu は Apache Hadoop プラットフォーム向けに開発された列指向ストレージです。Fuse Online のインテグレーションは Kudu データストアに接続してテーブルをスキャンでき、テーブルのすべてのレコードをインテグレーションに返すか、レコードをテーブルに挿入します。詳細は以下を参照してください。

- [「Apache Kudu データストアへの接続の作成」](#)
- [「スキャンが Kudu テーブルからレコードを返すときにインテグレーションをトリガー」](#)
- [「Kudu テーブルへのレコードの挿入」](#)

16.1. APACHE KUDU データストアへの接続の作成

インテグレーションで、Kudu テーブルからレコードを取得したり、レコードを Kudu テーブルに挿入するには、Kudu マスターホストへの接続を作成し、その接続をインテグレーションに追加します。

前提条件

- 接続する Kudu マスターホストの IP アドレスとホスト名を知っている必要があります。
- Kudu がリッスンするポートを知っている必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能な接続を表示します。
2. 右上の **Create Connection** をクリックしてコネクタを表示します。
3. **Apache Kudu** コネクタをクリックします。
4. 接続を設定します。
 - a. **Address of Kudu master host** フィールドに、Kudu マスターホストのホスト名または IP アドレスを入力します。
 - b. **Port to establish connection to** フィールドに、Kudu がリッスンするポートを入力します。デフォルトは **7051** です。
5. **Next** をクリックします。
6. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**Kudu North** を入力します。
7. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。
8. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **Kudu North** が表示されます。

16.2. スキャンが KUDU テーブルからレコードを返すときにインテグレーションをトリガー

Kudu テーブルからデータを取得したときにインテグレーションの実行をトリガーするには、Kudu コネクションを最初のコネクションとしてシンプルなインテグレーションに追加します。インテグレーションの実行時、Kudu コネクションは指定の間隔で指定のテーブルをスキャンし、レコードのコレクションをインテグレーションの次のステップに渡します。

Kudu コネクションは1つのテーブルからのみデータを取得できます。スキャンの合間に、コネクションがスキャンするテーブルのデータに変更がない場合は、次のスキャンが同じデータを前回のスキャンとして返します。

前提条件

- Kudu コネクションが作成されている必要があります。
- レコードの取得元となるテーブルが存在する必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する Kudu コネクションをクリックします。
4. **Choose an action** ページで **Scan a Kudu table** アクションを選択します。
5. **Table** フィールドに、レコードを取得するテーブルの名前を入力します。
6. **Period** フィールドで、デフォルトの1分を使用するか、コネクションがテーブルをスキャンし、テーブルのレコードを返す間隔を入力します。
7. **Next** をクリックします。

結果

コネクションはシンプルなインテグレーションの最初のコネクションです。

次のステップ

インテグレーションの最後のコネクションと、インテグレーションに含まれるようにする他のコネクションを追加します。インテグレーションに必要なすべてのコネクションが含まれる場合、Kudu コネクションが返すレコードのコレクションを分割する必要があるかどうか検討します。Kudu テーブルから取得した各レコードのインテグレーションステップを実行する場合、Kudu コネクションの後に分割ステップを追加します。また、インテグレーションの後続のコネクションで Kudu から取得したデータをフィールドにマップするデータマッピングステップで、Kudu コネクションにしたがう必要がある可能性が高くなります。

16.3. KUDU テーブルへのレコードの挿入

インテグレーションでは、フローの途中で Kudu テーブルにレコードを追加したり、シンプルなインテグレーションを終了するために Kudu テーブルにレコードを追加することができます。これには、Kudu コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Kudu コネクションが作成されている必要があります。
- インテグレーションを作成または編集することになります。Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。
- レコードの追加先となるテーブルが存在する必要があります。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. 使用する Kudu コネクションをクリックします。Kudu コネクションがデータを挿入するとき、コネクションは何も返さないことに注意してください。
3. **Choose an action** ページで **Insert data into a Kudu table** を選択します。
4. アクションを設定するには、**Table** フィールドでレコードの追加先となるテーブルの名前を指定します。
レコードの追加先となる Kudu テーブルの設定方法を理解することが重要になります。たとえば、レコードの追加先となる Kudu テーブルには一意のキーがある可能性があります。テーブルにすでに存在するキー値が含まれるレコードを追加しようとしても、Kudu コネクションはそのレコードを追加しません。
5. **Next** をクリックします。

結果

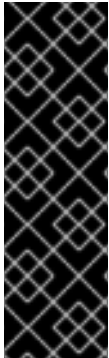
インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

次のステップ

レコードのコレクションを Kudu コネクションがテーブルに追加できる個別のレコードに分割する必要があるかどうかを検討します。これには、インテグレーションに分割ステップを追加します。分割ステップは、各レコードに対してインテグレーションの後続のステップを一度実行します。また、レコードとテーブルに追加する Kudu コネクションの前にデータマッピングステップが必要になる可能性が高いでしょう。

第17章 MONGODB への接続

MongoDB は分散型 NoSQL データベースです。Fuse Online インテグレーションは MongoDB データベースからコンテンツを取得したり、MongoDB データベースでコンテンツを更新したりできます。これには、MongoDB コネクションを作成し、インテグレーションに追加します。



重要

MongoDB への接続はテクノロジープレビューの機能です。テクノロジープレビュー機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、Red Hat は本番環境での使用は推奨しません。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能により、近日発表予定の製品機能をリリースに先駆けてご提供でき、お客様は開発プロセス時に機能をテストして、フィードバックをお寄せいただくことができます。Red Hat のテクノロジープレビュー機能のサポート範囲に関する詳細は、<https://access.redhat.com/ja/support/offerings/techpreview> を参照してください。

MongoDB データベースへの接続に関する詳細は、以下を参照してください。

- [「MongoDB データベースへの接続の作成」](#)
- [「MongoDB データベースの更新時にインテグレーションをトリガー」](#)
- [「MongoDB データベースのドキュメントの操作」](#)

17.1. MONGODB データベースへの接続の作成

MongoDB コネクションを作成して、インテグレーションで MongoDB データベースに接続できるようにします。

前提条件

- 接続する MongoDB データベースのホスト名を知っておく必要があります。
- MongoDB データベースへのアクセスが許可されるユーザー名とパスワードが必要です。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックしてコネクターを表示します。
3. **MongoDB** コネクターをクリックします。
4. コネクションを設定します。
 - a. **Database host(s)** フィールドに、このコネクションがアクセスできる MongoDB データベースホストを1つ以上指定します。複数の **host:port** の指定はコンマで区切ります。**host** はホスト名、仮想ホスト名、または IP アドレスに置き換えます。たとえば、**mongodb0.example.com:1234,mongodb1.example.com:1234** のように指定します。
 - b. **Username** フィールドに、データベースへのアクセスが許可されるユーザー名を入力します。

- c. **Password** フィールドに、指定したユーザー名のパスワードを入力します。
 - d. 任意手順:データベースに管理データベースがある場合は、**Administration database** フィールドにアクセス権限を付与するために使用される MongoDB 管理データベースの名前を入力します。アクセスするデータベースに管理データベースがあるかどうかを確認するには、MongoDB の管理者に問い合わせます。
 - e. 任意手順:コネクションがアクセスできる MongoDB レプリカセット (クラスター) の名前を **Replica set name** フィールドに指定します。
 - f. **Database** フィールドに、コネクションが読み取るコレクションが含まれる MongoDB データベースの名前を入力します。
5. **Validate** をクリックします。Fuse Online は即座にコネクションを検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、入力パラメーターを訂正し、再度検証を行います。
 6. 検証に成功したら **Next** をクリックします。
 7. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**MongoDB North** を入力します。
 8. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。
 9. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **MongoDB North** が表示されます。

次のステップ

MongoDB コネクションをインテグレーションに追加します。

17.2. MONGODB データベースの更新時にインテグレーションをトリガー

MongoDB データベースから更新を取得したときにインテグレーションの実行をトリガーするには、MongoDB コネクションを最初のコネクションとしてシンプルなインテグレーションに追加します。インテグレーションの稼働時、MongoDB コネクションはインテグレーションの作成時に指定したデータベースコレクションを監視します。監視されるコレクションは MongoDB の上限付きのコレクションである必要があります。コネクションが監視するように設定されたコレクションに新しいドキュメントを挿入すると、インテグレーションの実行がトリガーされます。MongoDB コネクションによってコレクションの更新がインテグレーションに返されます。このデータはインテグレーションの次のステップで使用できます。

前提条件

- MongoDB コネクションが作成されている必要があります。
- コネクションが読み取る MongoDB の上限付きコレクションの名前を知っておく必要があります。
- 受信ドキュメントを追跡する MongoDB コレクションフィールドの名前を知っておく必要があります。
- コネクションの読み取り元となるデータベースに対して永続トラッキングが有効になっているかどうかを知っておく必要があります。
- コネクションが読み取るデータに関連するテールトラッキング (Tail tacking) 情報の指定方法を

知っておく必要があります。MongoDB コネクションは、コネクションが予期せず終了した後、テールトラッキング情報を使用してアクションを再開します。テールトラッキングは、コネクションがアクティビティを再開したときに、重複したデータをインテグレーションに返さないようにしたり、インテグレーションに返す必要のあるデータを失わないようにします。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションが更新を取得するデータベースにアクセスするよう設定された、MongoDB コネクションをクリックします。
4. **Choose an action** ページで **Mongo consumer** アクションを選択します。
5. アクションを設定するには以下を行います。
 - a. **Collection name** フィールドに、コネクションが読み取る MongoDB コレクションの名前を入力します。
 - b. **Collection field used to track incoming documents** フィールドに、関連する MongoDB コレクションフィールドの名前を入力します。このフィールドには、日付や ID などのソート可能なデータが含まれます。
 - c. コネクションが読み取るデータベースの永続トラッキングが有効になっている場合は、**Persistent tracking enabled** チェックボックスを選択します。
 - d. 任意で、**Field used to store tracking progress** フィールドに、指定のコレクションにあるフィールドの名前を指定します。MongoDB 管理者がこのフィールドを設定済みである必要があります。
 - e. 任意で、**DB used to store tail tracking** フィールドに、テールトラッキング情報が含まれる MongoDB データベースの名前を入力します。
 - f. 任意で、**Collection used to store tail tracking** フィールドに、テールトラッキング情報が含まれるコレクションの名前を入力します。
 - g. 任意で、**Field used to store tail tracking** フィールドに、このコネクションが読み取るコレクションのテールトラッキング情報を保存するフィールドの名前を入力します。



注記

テールトラッキングフィールドを特定するために、テールトラッキングデータベースまたはテールトラッキングコレクションを指定できます。いずれかの情報を指定すると、コネクションはテールトラッキングフィールドを特定できます。

6. **Next** をクリックします。

結果

インテグレーションに最初のコネクションが追加され、Fuse Online は最後のコネクションを選択するよう要求します。

実行中、コネクションは挿入されたドキュメントを JSON 形式で返します。

17.3. MONGODB データベースのドキュメントの操作

インテグレーションでは、MongoDB データベースのドキュメントを取得、カウント、更新、および削除でき、MongoDB にドキュメントを追加することもできます。これには、MongoDB コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- MongoDB コネクションが作成されている必要があります。
- コネクションが操作する MongoDB コレクションの名前を知っておく必要があります。
- インテグレーションを作成または編集することになります。Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、MongoDB コネクションの追加先のプラス記号をクリックします。シンプルなインテグレーションの最後のコネクションを追加する場合は、このステップをスキップします。
2. 使用する MongoDB コネクションをクリックします。
3. **Choose an action** ページで **MongoDB producer** アクションを選択します。
4. アクションを設定するには以下を行います。
 - a. **Collection name** フィールドに、コネクションが操作するコレクションの名前を入力します。
 - b. **Operation to perform** フィールドで **Insert** 操作を許可するか、フィールドをクリックして利用可能なアクションを表示します。コネクションが実行するアクションをクリックします。各操作の説明は以下の表を参照してください。
各操作は応答を返します。これが最後のコネクションである場合、応答は失われます。そのため、これが最後のコネクションである場合、代わりに **Log** ステップを追加してインテグレーションを終了することが推奨されます。**Log** ステップを設定して応答をキャプチャーします。
5. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

次のステップ

すべてのコネクションをフローに追加します。データを処理する他のステップを追加します。このテクノロジープレビューリリースでは、ソースフィールドを MongoDB コネクションフィールドにマップすることはできません。そのため、コネクションに送信されるデータが、選択した操作に必要な入力を提供するようにする必要があります。今後のリリースで、データマッピングがサポートされる予定です。

表17.1 操作の説明

操作	動作の説明	入力	戻り値
Count	入力フィルター式を満たすドキュメントの数をカウントします。	<code>{"test":"unit"}</code> 形式の JSON フィルター式。 unit の test 属性値のあるドキュメントの数を数えます。	見つかったドキュメントの数を指定する JSON 式 (例: <code>{"count":2}</code>)。
Find by id	指定の ID を持つドキュメントを検索します。	対象の ID が含まれる文字列。	ID を持つドキュメントを指定する JSON 表記。
Find all	コレクションのすべてのドキュメントを検索します。	なし	コレクションにあるすべてのドキュメントを指定する JSON 表記。
Insert	コレクションにドキュメントを追加します。	コレクションに追加するドキュメントを指定する JSON 表記。	挿入されたドキュメントを指定する JSON 表記。
Remove	コレクションからドキュメントを削除します。	<code>{"test":"unit"}</code> 形式の JSON 式。test 属性に unit の値がある各ドキュメントを削除します。	削除されたドキュメントの数を指定する JSON 式 (例: <code>{"count":2}</code>)。
保存します。	ドキュメントをコレクションに追加、またはコレクションのドキュメントを更新します。	追加または更新するドキュメントを指定する JSON 表記。	保存したドキュメントを指定する JSON 表記。
Update	コレクションのドキュメントを更新します。	2つの引数の配列。最初の引数は、更新するドキュメントを識別するフィルター式です。2つ目の引数は、ドキュメントの更新方法を指定する MongoDB 更新ステートメントです。更新ステートメントの形式は、 https://docs.mongodb.com/manual/reference/method/db.collection.update/ に説明されています。(例: <code>[{"_id":11},{ \$set: {"test":"updated!"}]</code>)	更新されたドキュメントの数を指定する JSON 表記 (例: <code>{"count":2}</code>)。

第18章 MQTT への接続

MQTT (MQ Telemetry Transport) は、ライトウェイトなマシンツーマシンの IoT (Internet of Things、モノのインターネット) 接続プロトコルです。インテグレーションでは、MQTT ブローカーからメッセージを取得したり、MQTT へメッセージをパブリッシュすることができます。これには、対象の MQTT ブローカーへのコネクションを作成し、そのコネクションをインテグレーションフローに追加します。詳細は以下を参照してください。

- [「MQTT ブローカーへのコネクションの作成」](#)
- [「MQTT ブローカーからメッセージを取得してインテグレーションの実行をトリガー」](#)
- [「MQTT ブローカーへのメッセージの公開」](#)

18.1. MQTT ブローカーへのコネクションの作成

インテグレーションで、MQTT ブローカーからメッセージと取得したり、MQTT ブローカーにメッセージをパブリッシュするには、対象の MQTT ブローカーへのコネクションを作成し、そのコネクションをインテグレーションに追加します。

前提条件

接続する MQTT ブローカーの URL を知っている必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックしてコネクタを表示します。
3. **MQTT Message Broker** コネクタをクリックします。
4. コネクションを設定します。
 - a. **MQTT broker URL** フィールドにデータの送信先または取得元となる MQTT ブローカーの場所を入力します。必須のフィールドはこのフィールドのみです。
 - b. 任意で、このブローカーにアクセスするために使用するクレデンシャルを持つ MQTT アカウントのユーザー名を **User Name** フィールドに入力します。
 - c. ユーザー名を指定した場合、**Password** フィールドにそのアカウントに関連するパスワードを指定します。
 - d. 任意で、**Client ID** フィールドに、メッセージを見逃さずにコネクションを開閉できるようにする ID を入力します。コネクションには、トピックのサブスクライブまたはパブリッシュが必要です。
5. **Validate** をクリックします。Fuse Online は即座にコネクションを検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、入力パラメーターを訂正し、再度検証を行います。
6. 検証に成功した場合は **Next** をクリックします。
7. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**MQTT West** を入力します。

8. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。
9. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **MQTT West** が表示されます。

18.2. MQTT ブローカーからメッセージを取得してインテグレーションの実行をトリガー

MQTT ブローカーからのメッセージの受信により、インテグレーションの実行をトリガーするには、MQTT 接続を最初の接続として追加します。インテグレーションの実行中、MQTT 接続は指定した MQTT キューまたはトピックでメッセージを継続的に監視します。接続がメッセージを見つけると、そのメッセージをインテグレーションの次のステップに渡します。MQTT 接続は一度に1つのメッセージを処理します。

前提条件

MQTT 接続が作成されている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する MQTT 接続をクリックします。
4. **Choose an action** ページで、**Subscribe** アクションを選択し、指定したキューまたはトピックからメッセージを受信します。
5. **MQTT queue/topic name** フィールドに、データを受信するためにサブスクライブするキューまたはトピックの名前を入力します。
6. **Next** をクリックして、アクションの出力タイプを指定します。
7. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。
 - **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
8. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。

9. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときにこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
10. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
11. **Next** をクリックします。

結果

インテグレーションに最初のコネクションが追加され、Fuse Online は最後のコネクションを選択するよう要求します。

18.3. MQTT ブローカーへのメッセージの公開

インテグレーションでは、フローの途中で MQTT ブローカーにメッセージをパブリッシュしたり、シンプルなインテグレーションを終了するために MQTT ブローカーにメッセージをパブリッシュすることができます。これには、MQTT コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- MQTT コネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、MQTT コネクションの追加先のプラス記号をクリックします。シンプルなインテグレーションの最後のコネクションを追加する場合は、このステップをスキップします。
2. メッセージの公開に使用する MQTT コネクションをクリックします。
3. **Choose an action** ページで **Publish** を選択します。
4. **MQTT queue/topic name** フィールドに、メッセージを公開するキューまたはトピックの名前を指定します。
5. **Next** をクリックしてアクションの入力タイプを指定します。
6. データタイプの認識が必要がない場合は、**Select Type** フィールドに **Type specification not required** を使用し、**Next** をクリックします。残りの手順を行う必要はありません。これ以外の場合は、**Select Type** フィールドをクリックし、以下の1つをスキーマタイプとして選択します。
 - **JSON schema** は JSON データの構造を記述するドキュメントです。ドキュメントのメディアタイプは **application/schema+json** です。

- **JSON instance** は JSON データが含まれるドキュメントです。ドキュメントのメディアタイプは **application/json** です。
 - **XML schema** は XML データの構造を記述するドキュメントです。ドキュメントのファイル拡張子は **.xsd** です。
 - **XML instance** は XML データが含まれるドキュメントです。このドキュメントのファイル拡張子は **.xml** です。
7. **Definition** 入力ボックスに、選択したスキーマタイプに準拠する定義を貼り付けます。たとえば、**JSON schema** を選択した場合は、メディアタイプが **application/schema+json** の JSON スキーマファイルの内容を貼り付けます。
 8. **Data Type Name** フィールドにはデータタイプに選択した名前を入力します。たとえば、ベンダーの JSON スキーマを指定する場合は、**Vendor** をデータタイプ名として指定することができます。
このタイプを指定するコネクションを使用するインテグレーションを作成または編集するときこのデータタイプ名が表示されます。Fuse Online では、インテグレーションビジュアライゼーションとデータマッパーにタイプ名が表示されます。
 9. **Data Type Description** フィールドには、このタイプを区別するのに便利な情報を提供します。データマッパーでこのタイプを処理するステップにカーソルを合わせると、この説明が表示されます。
 10. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

第19章 ODATA への接続

OData (Open Data Protocol) は、REST API を構築および使用するための標準です。OData サービスは、OData 対応の HTTP リクエストを介して Fuse Online などのサービスをクライアントの提供します。Fuse Online インテグレーションは、OData サービスからエンティティを取得でき、OData サービスによって管理されるエンティティを更新、作成、または削除できます。これには、OData コネクションを作成し、そのコネクションをインテグレーションに追加します。

OData の仕様を理解していることを前提とします。

OData サービスへの接続に関する詳細は、以下を参照してください。

- 「OData サービスへのコネクションの作成」
- 「ポーリングが OData からデータを返すときにインテグレーションをトリガー」
- 「OData サービスによって管理されるデータの更新、作成、および削除」

19.1. ODATA サービスへのコネクションの作成

インテグレーションで、OData サービスによって管理されるエンティティを取得、更新、作成、または削除するには、最初に OData サービスへのコネクションを作成する必要があります。

前提条件

- 接続する OData サービスのベース URL を知っている必要があります。
- サービスが認証を使用する場合はクレデンシャルが必要で、必要に応じて SSL 証明書が必要です。しかし、ほとんどの OData サービスはパブリックサービスであり、認証は必要ありません。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックしてコネクタを表示します。
3. **OData** コネクタをクリックします。
4. コネクションを設定します。
 - a. **Service Root URL** フィールドに、アクセスする OData サービスのベース URL を入力します。
 - b. サービスが Basic 認証を使用する場合は、**User Name** フィールドにユーザー名を入力します。
 - c. サービスが Basic 認証を使用する場合は、**Password** フィールドにパスワードを入力します。
 - d. サービスが SSL 証明書を要求する場合は、**Server Certificate** フィールドに SSL 証明書の内容を貼り付けます。
通常、パブリックの OData サービスには、ブラウザーによって提供される以外の証明書は必要ありません。しかし、内部 OData サービスでは、独自の SSL 署名書を作成し、自己署名した可能性があります。この証明書は認証局によって信用されないため、OData サービ

スへの接続に失敗します。接続を有効にするには、ここに自己署名証明書を貼り付けます。

ブラウザでサービスをロードするのが証明書を取得する1つの方法です。その後のステップは、使用しているブラウザによって異なります。一般的に、アドレスバーの横にある **Not secure** 鍵マークをクリックして、**View Certificate** をクリックし、表示された証明書をファイルにエクスポートしてその証明書をコピーし、このフィールドに貼り付けるステップと似たものになります。

5. **Validate** をクリックします。Fuse Online は即座に接続を検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、入力パラメーターを訂正し、再度検証を行います。
6. 検証に成功したら **Next** をクリックします。
7. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**OData North** を入力します。
8. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。
9. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **OData North** が表示されます。

次のステップ

OData 接続をインテグレーションに追加します。

19.2. ポーリングが ODATA からデータを返すときにインテグレーションをトリガー

OData サービスからデータを取得したときにインテグレーションの実行をトリガーするには、OData 接続を最初の接続としてシンプルなインテグレーションに追加します。インテグレーションの実行時、OData 接続は指定された間隔でサービスをポーリングします。接続設定を満たすデータが見つかったら、接続はそのデータをインテグレーションの次のステップに渡します。

前提条件

OData 接続が作成されている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する OData 接続をクリックします。
4. **Choose an action** ページで **Read** アクションを選択します。
5. **Read** アクションを設定します。
 - a. **Resource Collection** フィールドで、クエリーを実行する OData リソースの名前を選択します。Fuse Online は OData サービスからデータを取得し、利用可能なリソースコレクションのリストを提供します。

- b. 特定の1つのエンティティを取得するには、**Entity Key Predicate** フィールドにキー述語を指定してそのエンティティを特定します。たとえば、**UserName='Bob'** や **Categories(1)** を指定します。複数のエンティティを取得する場合はこのフィールドを空白のままにしておきます。
- c. **Query Options** フィールドに、リソースに適用するクエリーを入力します。OData 構文を使用します。たとえば、**\$filter=startswith(name, 'N')** は、**N** で始まる **name** フィールドを持つリソースで各エンティティのメッセージを返します。
Entity Key Predicate と **Query Options** の両方を指定できます。両方を指定すると、OData サービスは指定されたエンティティを取得し、クエリーをそのエンティティに適用します。たとえば、**Entity Key Predicate** を **UserName='russellwhyte'/Emails** に設定し、**Query Options** を **\$filter=contains(\$it,'example')** に設定した場合、接続は以下のようなリクエストを作成します。

```
https://services.odata.org/TripPinRESTierService/People(UserName='russellwhyte')/Emails?&$filter=contains($it, 'example')
```

サービスは、ドメイン **example** が含まれる Russell Whyte のメールアドレスをすべて返します。

- d. **Filter Old Results** チェックボックスを選択し、特定のメッセージを一度のみ取得します。
- e. メッセージのコレクションでなく、個別のメッセージを接続が返すようにするには、**Split Results** チェックボックスを選択します。
Fuse Online は、フローに追加できる個別の分割ステップ (split step) および集約ステップ (aggregate step) も提供します。1つ以上のステップで個別のメッセージを処理し、メッセージをコレクションに集約する場合は、**Split Results** チェックボックスを選択しないでください。代わりに、このチェックボックスを空のままにし、この接続の後にフローに分割ステップ (split step) を追加します。フローに集約ステップ (aggregate step) を使用する場合は分割ステップ (split step) が必要になります。
- f. **Interval Before Polling Starts** フィールドには、デフォルトの1秒を使用するか、接続が OData サービスのポーリングを開始するまでの経過期間を入力します。
- g. **Delay** フィールドには、デフォルトの30秒を使用するか、接続が OData サービスをポーリングする間隔を入力します。
- h. **Backoff Idle Threshold** フィールドには、デフォルトの1を使用するか、戻りデータがなくても許容されるポーリングの連続数を示します。このポーリング数を超えると、接続は後続のポーリングの間隔を長くします。接続は、**Delay** の値に **Backoff Multiplier** の値を掛けて、ポーリングの新しい間隔を決定します。
たとえば、ポーリングの間隔 (**Delay** の値) がデフォルトの30秒である場合、**Backoff Idle Threshold** は5に設定され、**Backoff Multiplier** は12に設定されます。ポーリングが5回連続でデータを返さなかった場合、接続は360秒 (30 x 12) 待機してからポーリングを再開します。接続は、ポーリングがデータを返すまで、360秒間隔でポーリングを続行します。ポーリングがデータを返したら、接続はポーリング間隔を30秒に戻します。
- i. **Backoff Multiplier** フィールドには、デフォルトの1を使用するか、**Backoff Idle Threshold** に設定された値に達したときにポーリングの間隔を長くする乗数を整数で指定します。
Backoff Multiplier にデフォルトの1を使用すると、連続して結果を返さなかったポーリングの数に関わらず、接続は指定の間隔でポーリングを続行します。

コネクションはデータが返されないと自動的にポーリングの頻度を減らすことができるため、**Backoff Idle Threshold** および **Backoff Multiplier** に指定する値は、CPU オーバーヘッドの削減に便利です。

6. **Next** をクリックします。

結果

インテグレーションに最初のコネクションが追加され、Fuse Online は最後のコネクションを選択するよう要求します。

実行中にコネクションが返すものは **Entity Key Predicate** および **Query Options** フィールドに指定した内容によって異なります。OData コネクションは以下を返すことができます。

- エンティティのコレクションまたはエンティティプロパティのコレクション
たとえば、リソースのすべての **Person** エンティティであったり、すべての **Person** エンティティのすべての **Age** プロパティであったりします。コネクションはコレクションを1つのメッセージで返します。Fuse Online はコレクションに対して、フローで各後続ステップを1度実行します。しかし、**Read** アクションを設定するときに **Split Results** を選択すると、コネクションは独自のメッセージで各エンティティまたは各プロパティを返します。Fuse Online はフローで各後続ステップを一度各メッセージに対して実行します。
- エンティティまたはエンティティプロパティ
たとえば、**UserName** プロパティが **Bob** である **Person** エンティティであったり、**UserName** が **Bob** である **Person** エンティティの **Age** プロパティであったりします。コネクションは、フローの次のステップに渡すメッセージのエンティティまたはエンティティプロパティを返します。

次のステップ

インテグレーションの最後のコネクションと、インテグレーションに含まれるようにする他のコネクションを追加します。インテグレーションに必要なコネクションがすべて含まれ、OData コネクションがコレクションを返す場合、OData コネクションの後に分割ステップ (split step) を追加する必要があるか検討します。通常、インテグレーションは OData コネクションによって返されるデータをフローの後続のコネクションが使用できるフィールドにマップします。コレクションをマップすることもあります。より多くの場合でターゲットフィールドにマップするためにコレクションを分割する必要があります。

OData コネクションの後、フローにデータマッピングステップを追加します。フローでの正確な場所は、フローが実行することによって異なります。たとえば、OData コネクションの後にベーシックフィルタステップを追加してから、データマッピングステップを追加することがあります。

19.3. ODATA サービスによって管理されるデータの更新、作成、および削除

インテグレーションでは、フローの途中で OData サービスによって管理されるリソースを更新したり、シンプルなインテグレーションを終了するために OData サービスによって管理されるリソースを更新することができます。これには、OData コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- OData コネクションが作成されている必要があります。

- インテグレーションを作成または編集することになります。Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後の接続を選択するよう要求されます。

手順

1. **Add to Integration** ページで、OData 接続の追加先のプラス記号をクリックします。シンプルなインテグレーションの最後の接続を追加する場合は、このステップをスキップします。
2. 使用する OData 接続をクリックします。OData 接続がエンティティを更新または削除するときに、接続は何も返さないことに注意してください。OData 接続がエンティティを作成するとき、接続は新しいエンティティを返しません。
3. **Choose an action** ページで、接続が実行するアクションを選択します。
 - **Create** はエンティティを OData リソースに追加します。
 - **Delete** はエンティティを OData リソースから削除します。
 - **Update** は OData リソースにあるエンティティを変更します。
4. 更新する OData リソースを選択し、アクションを設定します。
更新する OData リソースの更新方法を理解することが重要になります。OData サービスのバックエンドデータソースは、更新や競合解決のルールを決定します。たとえば、OData 接続が新規エンティティを作成しようとしたにも拘らず、その Predicate Key が指定されたエンティティがすでに存在した場合に OData サービスは既存のエンティティを上書きするか、既存のエンティティのフィールドの一部を上書きするか、この操作を無視します。この状況で OData サービスがどのように動作するかを知るのはユーザー次第です。
5. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。

次のステップ

OData 接続の前にデータマッピングステップを追加します。エンティティを更新するために、新しいエンティティの作成に必要なデータを提供するソースフィールドをマップする必要があります。[インテグレーションデータを次の接続のフィールドにマッピング](#)を参照してください。

第20章 REST API への接続

インテグレーションで REST API に接続するには、API を説明する OpenAPI ドキュメントをアップロードして、API のコネクタを作成しておく必要があります。[Fuse Online でのアプリケーションの統合の REST API クライアントコネクタの追加および管理](#) を参照してください。

接続する REST API のコネクタが Fuse Online で利用可能である場合、REST API に接続するステップは次のとおりです。

- 「[Fuse Online を REST API クライアントとして登録](#)」 (必要な場合)
- 「[REST API クライアントコネクションの作成](#)」
- 「[API クライアントのインテグレーションへの追加](#)」

20.1. FUSE ONLINE を REST API クライアントとして登録


Fuse Online が REST API クライアントコネクタを作成する前に、API のセキュリティ要件を示すよう要求されます。OAuth プロトコルを使用する API では、Fuse Online がコネクタを作成すると、API のエントリーも Fuse Online の **Settings** ページに追加します。ここに Fuse Online の API へのアクセスを承認する API クライアント ID と API クライアントシークレットを提供します。

接続する API が OAuth を使用しない場合、このセクションを飛ばして [REST API クライアントコネクションの作成](#) を参照してください。

前提条件

接続する REST API の OAuth カスタムアプリケーション設定ページの URL を知っている必要があります。

手順

1. Fuse Online で以下を行います。
 - a. **Settings** ページで、上部付近にあるコールバック URL の右側で  をクリックし、Fuse Online 環境のコールバック URL をクリップボードにコピーします。この手順の後半でこの URL が必要になります。
 - b. 接続する API の名前を探し、その名前をクリックしてクライアント ID およびクライアントシークレットフィールドを表示します。
2. 別のブラウザウィンドウで、Fuse Online を接続する API の OAuth クライアントとして登録する必要があります。これを行う正確な手順は、API サービスごとに異なります。通常、API サービスは OAuth カスタムアプリケーション設定ページを提供します。そのページに移動し、以下を行います。
 - a. この手順の最初にコピーした Fuse Online のコールバック URL を提供します。
 - b. 入力が必要なその他のプロンプトに対応します。
 - c. API サービスが Fuse Online 環境に割り当てるクライアント ID およびクライアントシークレットを取得します。
3. 登録している API サービスの Fuse Online **Settings** ページエントリーに戻り、以下を行います。

- a. 割り当てられたクライアント ID を貼り付けます。
- b. 割り当てられたクライアントシークレットを貼り付けます。
- c. **Save** をクリックします。

20.2. REST API クライアント接続の作成

インテグレーションで REST API に接続するには、任意の数のインテグレーションを追加できる REST API への接続を作成します。

前提条件

- 接続する REST API のコネクタが作成されている必要があります。
- REST API が OAuth プロトコルを使用する場合、その REST API にアクセスできるクライアントアプリケーションとして Fuse Online 環境が登録されています。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能な接続を表示します。
2. 右上にある **Create Connection** をクリックし、利用できるコネクタを表示します。
3. 接続を作成する API のコネクタをクリックします。
4. プロンプトに回答し、追加情報を提供します。Fuse Online がプロンプトで要求する項目は API の定義によって決まります。
通常、API サービスにはセキュリティ要件があり、Fuse Online は OpenAPI ドキュメントが定義する特定のセキュリティタイプに応じたセキュリティの入力を要求します。セキュリティタイプは以下の1つになります。

セキュリティタイプ	Fuse Online の表示
HTTP Basic 認証	ユーザー名およびパスワードの入力フィールド。
OAuth 2.0	API 接続の登録クレデンシャルを Fuse Online が検証するためにクリックするボタン。これらのクレデンシャルはコネクタの作成時に提供されます。
API キー	API サービスから取得する必要がある API キーの入力フィールド。

5. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。
6. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。
7. **Save** をクリックし、作成した接続を選択でき、インテグレーションに追加できることを確認します。

20.3. API クライアントのインテグレーションへの追加

インテグレーションで REST API に接続するには、その REST API に接続するステップを追加します。

本リリースでは、REST API への接続をフローの途中のステップするか、シンプルなインテグレーションの最後の接続とすることができます。よって、REST API への接続を最初のコネクションとすることはできません。

前提条件

- REST API への接続が作成されている必要があります。
- フローを作成または編集することになります。
- フローに最初のコネクションがある必要があります。
- Fuse Online は最後の接続の選択またはインテグレーションの追加を要求します。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。Fuse Online が最後の接続を選択するよう要求する場合は、このステップをスキップします。
2. 使用できるステップを表示するページで、フローに追加する REST API コネクションをクリックします。
3. コネクションが実行するアクションを選択します。使用できるアクションは、Fuse Online にアップロードされ、接続先の API を説明する OpenAPI ドキュメントに指定されたリソース操作を基にします。
4. 選択したアクションに応じて、Fuse Online が要求するパラメーターを入力します。
5. **Next** をクリックします。

結果

インテグレーションビジュアライゼーションで追加した場所に接続が表示されます。

第21章 SALESFORCE への接続

インテグレーションで Salesforce に接続するには、Fuse Online 環境を Salesforce クライアントアプリケーションとして登録する必要があります。これにより、Fuse Online による Salesforce へのアクセスが許可されます。この登録が完了していると、Salesforce コネクションを作成でき、そのコネクションに任意の数のインテグレーションを追加することができます。詳細は以下を参照してください。

- [「Fuse Online を Salesforce クライアントアプリケーションとして登録」](#)
- [「Salesforce コネクションの作成」](#)
- [「Salesforce コネクションのインテグレーションへの追加」](#)

21.1. FUSE ONLINE を SALESFORCE クライアントアプリケーションとして登録


インテグレーションで Salesforce に接続するには、最初に Fuse Online 環境を Salesforce にアクセスできるクライアントアプリケーションとして登録する必要があります。これにより、Salesforce に接続するインテグレーションをいくつでも作成することができます。そのため、特定の Fuse Online 環境を1度だけ Salesforce に登録する必要があります。

Fuse Online を Salesforce クライアントアプリケーションとして登録できるのは、Fuse Online 環境ごとに1回のみです。しかし、各 Salesforce コネクションは同じ登録を使用しますが、異なるユーザークレデンシャルを使用することができます。

前提条件

Fuse Online にログインしている必要があります。Fuse Online にアクセスできる URL は、Red Hat Fuse Online Evaluation プログラムのウェルカムメールメッセージにあります。

手順

1. Fuse Online で以下を行います。
 - a. 左側のパネルで **Settings** をクリックします。
 - b. **Settings** ページで、上部付近にあるコールバック URL の右側で  をクリックし、Fuse Online 環境のコールバック URL をクリップボードにコピーします。この手順の最後の方でこの URL が必要になります。
 - c. **Salesforce** エントリーをクリックして **Client ID** および **Client Secret** フィールドを表示します。
2. 別のブラウザータブで、Salesforce アカウントにログインし、以下の手順にしたがって接続アプリケーションを作成します。これらの手順では、Salesforce Classic ユーザーインターフェイスを使用していることを仮定します。Salesforce Lightning Experience インターフェイスから切り替えるには、プロフィールアイコンをクリックして **Switch to Salesforce Classic** を選択します。追加情報は、Salesforce ドキュメントの [Create a Connected App](#) を参照してください。
 - a. Salesforce で右上にある **Setup** をクリックします。
 - b. 左側のパネルで **Build > Create > Apps** と選択します。
 - c. **Connected Apps** まで下にスクロールし、**New** をクリックします。

- d. 必要な情報を入力し、**Enable OAuth Settings** を選択します。
- e. **Callback URL** フィールドに、この手順の最初にコピーした Fuse Online URL を貼り付けます。たとえば、**https://app-proj912876.7b63.fuse-ignite.openshiftapps.com/api/v1/credentials/callback** などです。
- f. **OAuth Scopes** では以下を追加します。
 - **Access and manage your data (api)**
 - **Allow access to your unique identifier (openid)**
 - **Perform requests on your behalf at any time (refresh_token, offline_access)**
- g. **Configure ID token** を選択し **Include Standard Claims** を選択します。
- h. 下にスクロールし、**Save** をクリックします。
- i. 上にスクロールし、Salesforce が短い待機時間を示していることを確認します。

New Connected App

[Help for this Page](#) 

Allow from 2-10 minutes for your changes to take effect on the server before using the connected app.



- j. **Continue** をクリックします。
 - k. Salesforce が提供するコンシューマー鍵をコピーします。
3. Fuse Online の **Settings** ページに戻り、Salesforce が提供したコンシューマー鍵を Fuse Online の Salesforce **Client ID** フィールドに貼り付けます。
 4. Salesforce に戻り、Salesforce が提供するコンシューマーの秘密をコピーします。
 5. Fuse Online の **Settings** ページに戻り、Salesforce が提供したコンシューマーの秘密を Fuse Online の Salesforce **Client Secret** フィールドに貼り付けます。
 6. **Save** をクリックします。
 7. **Salesforce** エントリーをクリックして折りたたみます。

21.2. SALESFORCE コネクションの作成

Salesforce でデータにアクセスするインテグレーションを作成するには、最初に Salesforce コネクションを作成する必要があります。Salesforce コネクションの作成後、複数のインテグレーションで使用することができます。

前提条件

- Fuse Online にログインしている必要があります。Fuse Online にアクセスできる URL は、Red Hat Fuse Online Evaluation プログラムのウェルカムメールメッセージにあります。
- Fuse Online がウェブブラウザで開かれている必要があります。
- Fuse Online 環境が Salesforce にアクセスできるアプリケーションとして登録されている必要があります。

- Fuse Online の **Settings** ページで登録した後に受け取った Salesforce クライアント ID および クライアントシークレットが追加済みである必要があります。
Fuse Online インストールを Salesforce クライアントとして登録した後、必ず 2 - 10 分ほど待ってから Salesforce コネクションを作成してください。

手順

1. Fuse Online で、左側のパネルにある **Connections** をクリックして、使用できるコネクションを表示します。
2. 右上の **Create Connection** をクリックし、利用可能なコネクタを表示します。コネクタは、1つ以上のコネクションを作成するためのテンプレートです。
3. **Salesforce** コネクタをクリックします。
4. **Connect Salesforce** をクリックして、Salesforce の承認ページを表示します。承認ページを表示するには Salesforce にログインする必要があります。
Connect Salesforce が表示されない場合、Fuse Online 環境は Salesforce クライアントアプリケーションとして登録されていません。[Fuse Online を Salesforce クライアントアプリケーションとして登録](#) を参照してください。Salesforce コネクションの作成時に Fuse Online 環境が Salesforce クライアントアプリケーションとして登録されていない場合、Fuse Online は承認を要求する複数のフィールドを表示します。これらのフィールドに値を入力して Salesforce コネクションを作成することはできますが、推奨されません。



注記

以下のエラーは、Salesforce に正しい Fuse Online コールバック URL が指定されていないことを表しています。

error=redirect_uri_mismatch&error_description=redirect_uri%20must%20match%20configuration

このエラーメッセージが表示されたら、Salesforce で [Fuse Online を Salesforce クライアントアプリケーションとして登録](#) の手順にしたがって Fuse Online コールバック URL が指定されていることを確認してください。

5. **Allow** をクリックして、Fuse Online に戻ります。
6. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**SF Connect 1** を入力します。
7. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample Salesforce connection that uses my Salesforce login credentials.** を入力します。
8. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **SF Connect 1** が表示されます。

21.3. SALESFORCE コネクションのインテグレーションへの追加

インテグレーションでは、フローの途中で Salesforce に接続したり、シンプルなインテグレーションを開始または終了できます。これには、Salesforce コネクションをインテグレーションに追加します。

前提条件

- Salesforce コネクションが作成されている必要があります。
- インテグレーションを作成または更新することになります。
- Fuse Online は、最初または最後のコネクションの選択か、インテグレーションの追加を要求します。

手順

1. Salesforce コネクションをフローの途中に追加する場合は、そのコネクションを追加する場所のプラス記号をクリックします。Fuse Online が最初または最後のコネクションを要求する場合は、このステップをスキップします。
2. フローに追加する Salesforce コネクションをクリックします。インテグレーションが Salesforce に接続するために選択したコネクションを使用する場合、Fuse Online はそのコネクションに定義されたクレデンシャルを使用します。
3. 選択したコネクションが実行するアクションを選択します。フローに追加した各 Salesforce コネクションは選択したアクションのみを実行します。
4. アクションが操作する Salesforce オブジェクトを指定します。たとえば、コンタクト、リード、または価格表のエントリーなどがこれに該当します。**Object** フィールドをクリックして Salesforce オブジェクトのリストから選択するか、オブジェクトの名前を入力します。
5. **Next** をクリックして、コネクションをフローに追加します。

第22章 SAP CONCUR への接続

SAP Concur は、経費、旅費、および請求書を管理します。インテグレーションで SAP Concur に接続し、SAP Concur のさまざまなアクションの1つを実行できます。たとえば、SQL データベースにプロジェクトコードを保存して、経費をプロジェクトに関連付けることができるように、SAP Concur に同じプロジェクトコードが必要な場合に、SQL データベースを接続するインテグレーションを作成して、新規プロジェクトコードを取得するストアードプロシージャを実行できます。その後、インテグレーションは SAP Concur に接続し、新しいプロジェクトコードを該当のリストにアップロードできます。

以下のトピックを参照してください。

- [「SAP Concur へ接続するためのユーザーロール」](#)
- [「SAP Concur 実装サイト用の OAuth クレデンシャルの取得方法」](#)
- [「SAP Concur 本番サイト用の OAuth クレデンシャルの取得方法」](#)
- [「SAP Concur コネクターの設定」](#)
- [「SAP Concur コネクションの作成」](#)
- [「SAP Concur コネクションのインテグレーションへの追加」](#)
- [「インテグレーションデータフィールドを SAP Concur フィールドにマップする例」](#)

22.1. SAP CONCUR へ接続するためのユーザーロール

SAP Concur には、開発作業の実装サイトと実稼働で使用するための本番サイトがあります。実装サイトは本番サイトをシミュレートします。これにより、SAP Concur のサイトで実稼働で使用する前にインテグレーションを作成および改良することができます。

Fuse Online はいつでも SAP Concur 実装サイトまたは本番サイトに接続できます。よって、特定の Fuse Online 環境で両方の SAP Concur サイトへのコネクションを同時に行うことはサポートされません。各 Fuse Online 環境には、各 SAP Concur サイトにアクセスするための独自のクレデンシャルが必要で、実装サイトのクレデンシャルと本番サイトのクレデンシャルは異なります。

そのため、システム管理者と同等の権限を持つユーザーが以下を行うことが想定されています。

1. SAP Concur から OAuth クレデンシャルを 2 セット取得します。
 - 1 セットは SAP Concur 実装サイトにアクセスするためのクレデンシャル
 - もう 1 セットは SAP Concur 本番サイトにアクセスするためのクレデンシャル
2. 適切な SAP Concur サイトにアクセスするために、SAP Concur コネクターを設定します。
3. SAP Concur コネクションを作成します。

その後、ビジネスユーザーは SAP Concur コネクションを使用するインテグレーションを作成できます。

22.2. SAP CONCUR 実装サイト用の OAUTH クレデンシャルの取得方法

SAP Concur 実装サイトへアクセスするためのクレデンシャルを取得するには、SAP Concur に直接連絡し、以下をリクエストします。

- Fuse Online を SAP Concur 実装サイトへのアクセスが許可される新しいアプリケーションとして登録します。
- 実装サイトに新しいセットの OAuth クレデンシャルが必要です。
- クレデンシャルには **LIST** スコープが含まれている必要があります。
- 任意で、list アクション以外のアクションを実行する権限を取得するには、他の **スコープ** をリクエストします。
- これらのクレデンシャルの OAuth 付与タイプは **Authorization Grant Flow** である必要があります。
Postman などの REST クライアントを使用してリクエストを SAP Concur 実装サイトに送信するには、OAuth 付与タイプが **Password** のクレデンシャルが必要です。
- クレデンシャルが必要な環境の Fuse Online コールバック URL は次のようになります。
<https://syndesis.my-minishift.syndesis.io/api/v1/credentials/callback>
この URL の **syndesis.my-minishift.syndesis.io** は、Fuse Online 環境への OpenShift ルートと一致します。

SAP Concur からの応答は以下を提供する必要があります。

- SAP Concur 実装サイトへログインするためのユーザー名およびパスワード。
- Fuse Online の SAP Concur コネクターを設定するために Fuse Online 開発環境の **Settings** に指定するクライアント ID およびクライアントシークレットの値。

22.3. SAP CONCUR 本番サイト用の OAUTH クレデンシャルの取得方法

Fuse Online の SAP Concur 本番サイトへのアクセスを許可する OAuth クレデンシャルを取得するには、SAP Concur Developer Center の [Getting Started](#) ページにアクセスし、記載されている手順にしたがいます。

22.4. SAP CONCUR コネクターの設定

インテグレーションで SAP Concur に接続するには、Fuse Online の SAP Concur コネクターを設定する必要があります。設定後、コネクターを使用して SAP Concur へのコネクションを作成できます。

前提条件

- SAP Concur コネクターを設定して SAP Concur 実装サイトまたは本番サイトにアクセスするかどうかを認識している必要があります。
- Fuse Online の適切な SAP Concur サイトへのアクセスを許可する OAuth クレデンシャルを取得している必要があります。

手順

1. Fuse Online の左側のパネルで **Settings** をクリックします。
2. **Settings** ページの **SAP Concur** エントリーを展開します。
3. **Client ID** フィールドに、SAP Concur から受け取ったクライアント ID を入力します。

4. **Client Secret** フィールドに、SAP Concur から受け取ったクライアントシークレットを入力します。
他のフィールドは Fuse Online によって入力されます。
5. **Save** をクリックします。
Fuse Online は即座に設定の検証を試みます。検証に成功しなかった場合は、入力値を修正して再度検証を行います。検証に成功した場合は、インテグレーションに追加できる SAP Concur コネクションを作成することができます。
6. **SAP Concur** エントリーをクリックして折りたたみます。

次のステップ

SAP Concur コネクションを作成します。

22.5. SAP CONCUR コネクションの作成

インテグレーションで SAP Concur に接続するには、任意の数のインテグレーションを追加できる SAP Concur コネクションを最初に作成する必要があります。SAP Concur コネクションを作成するとき、コネクションの作成時に入力するログインクレデンシャルを使用してコネクションが SAP Concur にアクセスすることを許可します。

前提条件

- コネクションを作成する Fuse Online 環境で、SAP Concur コネクターを設定する必要があります。
- SAP Concur 実装サイトまたは SAP Concur 本番サイトへアクセスするためにコネクターが設定されたかどうかを認識する必要があります。
- SAP Concur コネクターが設定された SAP Concur サイトへログインするためのユーザー名およびパスワードが必要です。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックし、利用可能なコネクターを表示します。コネクターは、1つ以上のコネクションを作成するために使用するテンプレートです。
3. **SAP Concur** コネクターをクリックします。
4. **Configure Connection** ページで **Connect SAP Concur** をクリックし、コネクターが設定された SAP Concur サイトに移動します。
Connect SAP Concur が表示されない場合は、Fuse Online 環境で SAP Concur コネクターが設定されていません。[SAP Concur コネクターの設定](#) を参照してください。

redirect_uri is improper or not previously registered が表示された場合、SAP Concur コネクター設定は正しくありません。Fuse Online の **Settings** ページに戻りコネクター設定を更新および検証します。

5. SAP Concur の OAuth サイトで以下を行います。
 - a. SAP Concur へアクセスするためにこのコネクションが使用する SAP Concur アカウントのメールアドレスを入力します。

- b. **Username** が選択されていることを確認してください。
 - c. **Continue** をクリックします。
 - d. このメールアドレスに関連する SAP Concur パスワードを入力します。
 - e. **Sign In** をクリックし、Fuse Online に戻ります。
6. Fuse Online の **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**SAP Concur Test1** を入力します。
 7. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample SAP Concur connection to the implementation site** を入力します。
 8. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **SAP Concur Test1** が表示されます。

22.6. SAP CONCUR コネクションのインテグレーションへの追加

SAP Concur コネクションをフローの途中に追加したり、シンプルなインテグレーションの最後のコネクションとして追加することができます。SAP Concur へのコネクションは 85 個のアクションのいずれかを実行できます。

前提条件

- SAP Concur コネクションが設定されている必要があります。
- フローを作成または編集することになります。
- Fuse Online はインテグレーションの追加を要求したり、シンプルなインテグレーションの最後のコネクションを選択するよう要求します。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。
2. フローに追加する SAP Concur コネクションをクリックします。選択したコネクションをインテグレーションが使用する場合、Fuse Online はそのコネクションに定義されたクレデンシャルを使用して SAP Concur に接続します。
3. コネクションが実行するアクションを選択します。フローに追加した各 SAP Concur コネクションは選択したアクションのみを実行します。
SAP Concur アクションに関する詳細は、[SAP Concur developer center](#) にアクセスし、**v3.0** を展開してください。
4. **Next** をクリックして、コネクションをフローに追加します。

結果

インテグレーションビジュアライゼーションで追加した場所にコネクションが表示されます。

22.7. インテグレーションデータフィールドを SAP CONCUR フィールドにマップする例

ほとんどのインテグレーションでは、SAP Concur への接続の前にデータマッピングステップを追加する必要があります。データマッピングステップでは、以前のインテグレーションステップから利用可能なデータフィールドを SAP Concur アクションパラメーターにマップします。データマッパーでは、インテグレーションに SAP Concur コネクションを追加したときに選択したアクションの SAP Concur パラメーターが **Target** パネルに表示されます。

新しいプロジェクトコードを取得する SQL ストアドプロシージャを実行してインテグレーションが起動するとします。インテグレーションは、新しいプロジェクトコードを SAP Concur に追加して終了します。以下の手順は、SAP Concur コネクションの前のデータマッピングステップの例を提供します。インテグレーションビジュアライゼーションが表示され、Fuse Online はインテグレーションの追加を要求します。

1. インテグレーションビジュアライゼーションで、SAP Concur の最後の接続の前にあるプラス記号をクリックします。
2. **Data Mapper** をクリックします。データフィールドが表示されると、左側の **Sources** パネルにデータベース接続から利用できるフィールドが表示されます。この例では、ソースフィールドに **concur-list-id** および **project-code** が含まれます。右側の **Target** パネルには、接続が実行する update list アクションの SAP Concur **id** および **content** パラメーターが表示されます。
3. **concur-list-id** ソースフィールドを **id** ターゲットフィールドにマップします。
4. **project-code** ソースフィールドを **content** ターゲットフィールドにマップします。
5. 右上の **Done** をクリックし、データマッピングステップをインテグレーションに追加します。

第23章 SERVICENOW への接続

インテグレーションは ServiceNow テーブルからレコードを取得したり、ServiceNow がテーブルを更新するために使用する ServiceNow インポートセットにレコードを追加することができます。インテグレーションで ServiceNow に接続するには、ServiceNow コネクションを作成した後、そのコネクションをインテグレーションに追加します。詳細は、以下を参照してください。

- 「ServiceNow コネクションの作成」
- 「ServiceNow からレコードを取得してインテグレーションの実行をトリガー」
- 「ServiceNow でのインポートセットの作成」
- 「インテグレーションの間またはインテグレーションを終了するためにレコードを ServiceNow にコピー」
- 「Salesforce から ServiceNow へのインテグレーションの例」

23.1. SERVICENOW コネクションの作成

インテグレーションで、所属企業の ServiceNow インスタンスに接続するには、ServiceNow コネクションを作成する必要があります。

前提条件

- 所属企業の ServiceNow 管理者がユーザーの ServiceNow アカウントを作成済みである必要があります。
- ServiceNow インスタンスの URL と、ServiceNow のユーザー名およびパスワードを知っている必要があります。
- ServiceNow 管理者が ServiceNow クライアント ID およびクライアントシークレットも作成した場合、これらの値を知っている必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクターを表示します。
3. **ServiceNow** コネクターをクリックします。
4. この ServiceNow コネクションを設定します。
 - a. **Instance Name** フィールドに、レコードの取得先または取得元となる ServiceNow インスタンスの名前を入力します。たとえば、ServiceNow インスタンスの URL が **https://abc12345.service-now.com** である場合、インスタンス名は **abc12345** になります。
 - b. **User Name** フィールドに ServiceNow ユーザー名を入力します。
 - c. **User Password** フィールドに ServiceNow パスワードを入力します。
 - d. 任意で、ServiceNow 管理者からクライアント ID を受け取った場合は **Client ID** フィールドにそのクライアント ID を入力します。

- e. 任意で、ServiceNow 管理者からクライアントシークレットを受け取った場合は **Client Secret** フィールドにそのクライアントシークレットを入力します。
 - f. **Validate** をクリックします。Fuse Online は即座に接続を検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を更新し、再度検証を行います。
 - g. 検証に成功したら **Next** をクリックします。
5. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**ServiceNow Con 1** を入力します。
 6. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。たとえば、**ServiceNow connection that uses administrative credentials.** を入力します。
 7. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **ServiceNow Con 1** が表示されます。

23.2. SERVICENOW からレコードを取得してインテグレーションの実行をトリガー

ServiceNow からレコードを受信したときにインテグレーションの実行をトリガーするには、ServiceNow 接続を最初の接続としてシンプルなインテグレーションに追加します。

前提条件

- ServiceNow 接続が作成されている必要があります。
- レコードを取得するテーブルの名前を知っている必要があります。
- 必要なレコードのみを取得する ServiceNow クエリーがすでに定義されている必要があります。ServiceNow クエリーの詳細は、ServiceNow の [Encoded query strings](#) を参照してください。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用する ServiceNow 接続をクリックします。
4. **Choose an action** ページで **Retrieve Record** を選択し、選択した ServiceNow テーブルからレコードを取得します。
5. **Retrieve Record** アクションを設定します。
 - a. **Table name** をクリックし、レコードを取得するテーブルを選択します。
 - b. **Query to filter the result set** フィールドに ServiceNow クエリーを入力します。たとえば、インシデントに関するレコードを保持するテーブルでの **state=1^impact=2** クエリーについて考えてみましょう。このクエリーは中程度の影響 (**impact=2**) を持つ新しいインシデント (**state=1**) のレコードを返します。クエリーを入力せず、テーブルのレコードが変更しないと、接続は同じレコードを毎回取得します。

- c. **Limit of elements per page** フィールドに、コネクションが取得する最大レコード数を入力します。本リリースでは、値を入力する必要があり、**1000** 以下の値を入力することが推奨されます。ページネーションはサポートされません。
- d. **Period** フィールドにレコードを取得する頻度を指定します。デフォルト毎分です。そのため、デフォルトでは Fuse Online はこのインテグレーションを 60 秒ごとに実行します。
- e. **Next** をクリックします。

結果

Fuse Online は、取得したレコードの構造を定義する JSON スキーマを生成します。これにより、データマッピングが必要な場合にインテグレーションで次のコネクションの前にデータマッピングステップを追加することができます。

Fuse Online は最後のコネクションの選択を要求します。

23.3. SERVICENOW でのインポートセットの作成

Fuse Online インテグレーションでは、ServiceNow コネクションは直接 ServiceNow テーブルを更新できません。インテグレーションで ServiceNow データを更新するには、ServiceNow テーブルの更新をステージングするインポートセットが事前に必要になります。

多くの場合で、ServiceNow 管理者が ServiceNow インバウンド Web サービスを作成することが、最も簡単にインポートセットを作成する方法になります。これにより、ServiceNow インポートセットが自動的に作成されます。インポートセットは、Web サービスの作成時に管理者が選択する ServiceNow ターゲットテーブルを基にしています。ServiceNow コネクションはこのインポートセットを更新し、ServiceNow はインポートセットを使用して対応する ServiceNow テーブルを更新します。

前提条件

ServiceNow の管理者権限が必要です。

ServiceNow の手順

1. ServiceNow で **web services** を検索します。
2. 結果の **System Web Services > Inbound** を選択し、**Create New** をクリックします。
3. **Create Web Service** ページで以下を行います。
 - a. **Label** フィールドで Web サービスの表示名を指定します。Web サービスのインポートセットにもこの名前が使用されます。
 - b. **Target table** フィールドに、このサービスが更新する ServiceNow テーブルを選択します。
 - c. **Copy fields from target table** を選択します。これにより、ターゲットテーブルと同じスキーマを持つインポートセットが作成されます。インポートセットはターゲットテーブルを更新するためのステージングエリアです。
 - d. **Create transform map** を選択します。このマップによって、ServiceNow がインポートセットからターゲットテーブルにデータをコピーまたは変換することができます。
4. **Create** をクリックします。

5. 次の表示の **Related Links** で **Auto Map Matching Fields** をクリックし、選択したターゲットテーブルにあるフィールドのリストを表示します。
6. **correlation** (相関) または **coalesce** (結合) フィールドとなるターゲットテーブルフィールドを1つ選択します。
インポートセットのレコードにある **correlation** フィールドの値が、ターゲットテーブルの **correlation** フィールドの値と一致する場合、ServiceNow は新しいレコードを作成せずに、ターゲットテーブルのレコードを更新します。
7. **Coalesce** を選択し、選択したフィールドを **correlation** フィールドとして特定します。
correlation フィールドがない場合、ServiceNow はインポートセットの各レコードをターゲットテーブルとして追加します。
8. **Update** をクリックします。
9. 変換マップ (transform map) が作成されたことを確認します。
 - a. ServiceNow で **transform map** を検索します。
 - b. **System Import Sets** で **Transform Maps** をクリックし、変換マップのリストを表示します。
 - c. このリストで、作成した Web サービスのラベルに指定した値が名前になっている変換マップがあることを確認します。

その他のリソース

ServiceNow ドキュメントの [Import sets](#)

23.4. インテグレーションの間またはインテグレーションを終了するためにレコードを **SERVICENOW** にコピー

フローの途中でレコードを ServiceNow にコピーしたり、シンプルなインテグレーションを終了するためにレコードを ServiceNow にコピーすることができます。これには、ServiceNow コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- ServiceNow コネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。
- レコードの追加先となる ServiceNow インポートセットの名前を知っている必要があります。ServiceNow 管理者は適切なインポートセットを特定することができます。インポートセットが存在しない場合は、[ServiceNow でのインポートセットの作成](#) を参照してください。
- 追加のレコードを処理するよう ServiceNow インポートセットが設定されている必要があります。

手順


1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。

2. インテグレーションを追加する ServiceNow コネクションをクリックします。
3. **Add Record** アクションを選択します。
4. **Import Set** フィールドをクリックし、レコードに追加するインポートセットを選択します。
5. **Next** をクリックします。

結果

Fuse Online はコネクションをフローに追加します。

通常、このコネクションの前にデータマッピングステップが必要になります。インテグレーションビジュ

アライゼーションで、Fuse Online はコネクション上に  を表示します。ほとんどのインテグレーションでは、ソースフィールドを ServiceNow テーブルの correlation フィールドとして使用されるターゲットフィールドにマップする必要があります。

23.5. SALESFORCE から SERVICENOW へのインテグレーションの例

この例では、2つのシンプルなインテグレーションについて説明します。

- 1つのインテグレーションは、Salesforce から新しいケースを取得し、インシデントとして ServiceNow に追加します。
- もう1つのインテグレーションでは、Salesforce から更新されたケースを取得し、ServiceNow の対応するインシデントを更新します。

ServiceNow における前提条件

両方のインテグレーションの前提条件として、ServiceNow の管理者がインバウンド Web サービスを作成する必要があります。これは、インテグレーションがデータを ServiceNow に追加するために必要な ServiceNow インポートセットを暗黙的に作成します。この例では、以下の条件が適用されます。

- Web サービスの名前は **Cases from Salesforce** です。
- サービスのターゲットテーブルは **Incidents** です。このテーブルには更新するデータが含まれます。
- **Copy fields from target table**が選択されているため、インポートセットには **Incidents** テーブルと同じスキーマがあります。インポートセットの名前も **Cases from Salesforce** になります。
- **Create transform map**が選択されているため、ServiceNow は **Cases from Salesforce** インポートセットを **Incidents** ターゲットテーブルにコピーおよび変換します。
- **correlation_Id** フィールドが選択され、その後に **Coalesce** が選択されているため、インポートセットでは correlation フィールドは **correlation_Id** になります。

Fuse Online での手順の概要

Fuse Online で Salesforce および ServiceNow コネクションを作成します。

以下のステップのとおり、2つのインテグレーションは、Salesforce コネクションで実行されるアクション以外は同じになります。インテグレーションを作成するには、これらのステップを2回実行し、1回目は各インテグレーションを作成します。

1. Salesforce コネクションを最初のコネクションとして追加します。

- 新しいケースを取得するインテグレーションを作成するには、**On create** アクションを選択します。
 - 更新されたケースを取得するインテグレーションを作成するには、**On update** アクションを選択します。
2. アクションが操作する **Case** オブジェクトを選択します。この Salesforce コネクションは、選択したアクションに応じて、新しい **Case** オブジェクトまたは更新された **Case** オブジェクトを返します。
 3. ServiceNow コネクションを最後のコネクションとして追加します。
 4. **Create Record** アクションを選択した後、インテグレーションが更新するインポートセットとして **Cases from Salesforce** を選択します。
この ServiceNow コネクションは、新しいまたは更新された Salesforce ケースを **Cases from Salesforce** インポートセットに追加します。
 5. 以下をマップするデータマップステップを追加します。
 - Salesforce ケース **id** から ServiceNow インポートセットの **correlation_id**
 - Salesforce **subject** から ServiceNow **short_description**
 - Salesforce **description** から ServiceNow **description**
 6. インテグレーションに名前を付け、公開します。
 - 新しいケースを取得するインテグレーションでは、**On SF Create Case** を使用します。
 - 更新されたケースを取得するインテグレーションでは、**On SF Update Case** を使用します。

インテグレーションの動作確認

両方のインテグレーションが実行しているときに、インテグレーションの動作を確認することができます。

1. Salesforce でケースを作成します。
2. Fuse Online で **On SF Create Case** インテグレーションの概要を表示します。**Activity** タブをクリックし、Fuse Online がインテグレーションを1度実行したことを確認します。
3. ServiceNow で、**Incidents** テーブルを表示します。Salesforce で指定した件名と説明のある新しいインシデントが表示されるはずですが、
4. Salesforce に戻り、作成したケースの件名を変更して、そのケースを更新します。
5. Fuse Online で **On SF Update Case** インテグレーションの概要を表示します。**Activity** タブをクリックし、Fuse Online がインテグレーションを1度実行したことを確認します。
6. ServiceNow で **Incidents** テーブルを表示し、これまで新規だったインシデントのエントリを展開します。このインシデントに更新された **short_description** の値があることが確認できるはずですが、ServiceNow は、**correlation_id** の値のインポートセットエントリをチェックします。この値がすでに **Incidents** テーブルに存在する場合、ServiceNow はその値があるインシデントを更新します。

第24章 SLACK への接続

ビジネスユーザーは Slack に接続するインテグレーションを作成できます。Slack への接続は以下の1つを行うことができます。

- 指定の Slack チャンネルがメッセージを受信するときにシンプルなインテグレーションの実行をトリガーします。インテグレーションはメッセージをフローの次のステップに渡します。たとえば、製品名などのキーワードインスタンスを Slack チャンネルで監視することができます。これらの製品名が含まれるメッセージを見つけた後、インテグレーションは Gmail 接続の適切な連絡先に通知することができます。
- メッセージを特定ユーザーまたはチャンネルに配信します。たとえば、この動作は、インテグレーションが FTP サーバーからファイルをダウンロードし、何らかの方法で処理する場合に便利です。インテグレーションフローは、プロセスに成功したことを Slack チャンネルまたはユーザーに通知できます。

インテグレーションで Slack に接続するには、Slack 接続を作成した後に接続をインテグレーションフローに追加します。詳細は以下を参照してください。

- [「Slack 接続の作成」](#)
- [「メッセージの受信時に Slack 接続を追加してインテグレーションの実行をトリガー」](#)
- [「Slack 接続を追加してメッセージを Slack チャンネルまたはユーザーに送信」](#)

24.1. SLACK 接続の作成

インテグレーションでは、Slack 接続は指定したチャンネルからメッセージを取得したり、メッセージをチャンネルまたはユーザーに送信することができます。同じ Slack 接続を任意の数のインテグレーションで使用することができます。

前提条件

- Fuse Online を Slack に接続する Slack アプリケーションが作成されている必要があります。
- Slack アプリケーションの Slack Webhook URL を取得できる必要があります。
- Slack アプリへのアクセスを許可するレガシートークンを取得できる必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能な接続を表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクタを表示します。
3. **Slack** コネクタをクリックします。
4. **Slack webhook URL** フィールドに Slack アプリケーションに Webhook URL を入力します。
5. **Token for accessing Slack API** フィールドに、[Slack から取得](#) できる Slack 提供のトークンを入力します。
6. 任意で、追加パラメーターの値を入力します。

- **Sending messages for messages** フィールドに、Slack からメッセージを送信するときに

- **Sending username for messages** フィールドに、Slack ハメッセンを返信するときにボットが持っているユーザー名を入力します。
- **Message avatar emoji** フィールドに、ボットがメッセージの送信時にメッセージアバターとして使用する絵文字を1つ指定します(例: **:smile:**、**:wave:**、**:question:**)。

ヒント

絵文字コードについては、[emoji cheat sheet](#) を Web 検索してください。

- **Message avatar icon URL** フィールドに、ボットがメッセージを Slack に送信するときに使用するアバターの URL を指定します。

絵文字とアイコン URL を指定した場合、インテグレーションはアイコン URL を使用します。絵文字とアイコン URL を両方指定しなかった場合、メッセージはアバターなしで送信されます。

7. **Validate** をクリックします。Fuse Online は即座に接続を検証しようとし、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、接続設定値を修正し、再度検証を行います。
8. 検証に成功した場合は **Next** をクリックします。
9. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**Slack for Tracking Company Sales** を入力します。
10. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。
11. **Save** をクリックし、作成したばかりの接続を含む、利用可能な接続の更新済みリストを表示します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **Slack for Tracking Company Sales** が表示されます。この接続を任意の数の接続に追加できます。

24.2. メッセージの受信時に SLACK 接続を追加してインテグレーションの実行をトリガー

指定した Slack チャンネルで接続がメッセージを見つけると、インテグレーションを開始する Slack 接続はインテグレーションの実行をトリガーします。

前提条件

- Slack 接続が作成されている必要があります。
- インテグレーションを作成することになります。Fuse Online はインテグレーションの開始方法の選択を要求します。

手順

1. インテグレーションの開始に使用する Slack 接続をクリックします。
2. **Read Messages** アクションを選択します。
3. **Channel** フィールドに、メッセージを取得する Slack チャンネルを指定します。

4. **Delay** フィールドで、デフォルトの 500 ミリ秒をメッセージのポーリング間の経過時間として受け入れます。別の間隔でメッセージをチェックする場合は、数値を入力し、時間の単位を選択します。
5. **Maximum Messages to Retrieve** フィールドには、デフォルトの 10 を使用するか、各ポーリングで接続が返すことができるメッセージの最大数を指定します。接続がこの値を超える数の未読メールを見つけた場合、最も新しいメッセージを返します。接続は特定のメッセージを1度のみ返します。
たとえば、デフォルトの 10 を使用し、ポーリング時に 15 個のメッセージがあった場合、接続は新しい順に 10 個のメッセージを返します。次のポーリングでは、接続が返した最新のメッセージよりも新しいタイムスタンプを持つメッセージをチェックします。そのため、最初のポーリングで返されなかった 5 個のメッセージは永遠に返されません。

Slack 接続が 1 つ以上のメッセージを返すとき、インテグレーションはメッセージをバッチとして処理します。つまり、Fuse Online はバッチに対してインテグレーションを 1 度実行します。
6. **Next** をクリックして接続をインテグレーションに追加します。

24.3. SLACK 接続を追加してメッセージを SLACK チャネルまたはユーザーに送信

インテグレーションでは、フローの途中でメッセージを Slack チャネルに送信したり、シンプルなインテグレーションを終了するためにメッセージを Slack チャネルに送信することができます。これには、Slack 接続をフローの途中で追加するか、シンプルなインテグレーションの最後の接続として追加します。

前提条件

- Slack 接続が作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後の接続を選択するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。Fuse Online が最後の接続を選択するよう要求する場合は、このステップをスキップします。
2. フローに追加する Slack 接続をクリックします。
3. 接続が実行するアクションを選択します。
 - **Username** を選択し、メッセージを 1 名のユーザーに送信します。このアクションを設定するには、**User name** フィールドにメッセージの送信先となるユーザーの Slack 名を指定します。
 - **Channel** を選択して、チャネルにメッセージを公開します。このアクションを設定するには、**Channel** フィールドにメッセージを公開するチャネルを指定します。
4. **Next** をクリックして、接続をフローに追加します。
5. 任意で、追加の接続をフローに追加します。追加の接続が必要であるかどうかは、フローが何を行うかによって異なります。すべての接続を追加してから手順を続行することが重要になります。

6. この手順を追加する Slack コネクションの直前にデータマッパーステップを追加します。マッピングステップでは、文字列をデータマッピングソースから Slack **message** フィールドにマップします。この文字列には、Slack ユーザーまたはチャンネルに送信するメッセージが含まれる必要があります。[データマッパーステップの追加](#) を参照してください。

第25章 SQL データベースへの接続

インテグレーションでは、以下の SQL データベースに接続することができます。

- Apache Derby
- MySQL
- PostgreSQL

インテグレーションでアクセスするデータベースへのコネクションを作成します。その後、インテグレーションを作成し、データベースコネクションをインテグレーションに追加します。

他の種類のデータベースに接続するには、そのデータベースの JDBC ドライバーをアップロードする必要があります。

詳細は以下のトピックを参照してください。

- [「データベースコネクションの作成」](#)
- [「データベースコネクションでスキーマを指定する方法」](#)
- [「データベースレコードを取得してインテグレーションの実行をトリガー」](#)
- [「データを更新する SQL ステートメントのパラメータープレースホルダーおよび値」](#)
- [「インテグレーションの途中またはインテグレーションを完了するためにデータベースにアクセス」](#)
- [「プロプライエタリーデータベースへの接続」](#)

25.1. データベースコネクションの作成

インテグレーションで接続する各データベースの個別のコネクションを作成します。複数のインテグレーションで同じコネクションを使用できます。

データベースコネクションは指定のデータベーステーブルで操作するか、指定のストアードプロシージャを呼び出します。

前提条件

インテグレーションがデータベースに接続するときにデータベーステーブルまたはストアードプロシージャが存在する必要があります。

手順

1. 接続するデータベースの JDBC ドライバーがクラスパスにあるようにしてください。プロプライエタリーデータベースに接続するために JDBC ドライバーライブラリーエクステンションをアップロードした場合、ドライバーはアップロードプロセスによってクラスパス上に置かれます。[JDBC ドライバーライブラリーエクステンションの作成](#) を参照してください。
2. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
3. 右上にある **Create Connection** をクリックし、Fuse Online コネクターを表示します。
4. **Database** コネクターをクリックします。

5. 以下を入力して接続を設定します。
 - a. **Connection URL** フィールドに、接続するデータベースの JDBC URL を入力します。たとえば、**jdbc:postgresql://ignite-db1234/sampledb** を入力します。
 - b. **Username** フィールドに、データベースのアクセスに使用するアカウントの名前を入力します。同じアカウントのパスワードとユーザー名を指定するようにしてください。
 - c. **Password** フィールドに、データベースのアクセスに使用する、ユーザーアカウントに関連するパスワードを入力します。
 - d. **Schema** フィールドにデータベースのスキーマの名前を入力します。データベーススキーマの指定方法は、データベースのタイプによって異なります。詳細は、次のセクションの [データベース接続でスキーマを指定する方法](#) を参照してください。
6. **Validate** をクリックします。Fuse Online は接続の検証を試み、検証の成功または失敗を示すメッセージを表示します。検証に失敗した場合は、必要に応じて設定の詳細を修正し、再度検証を行います。
7. 検証に成功した場合は **Next** をクリックします。
8. **Name** フィールドに、この接続を別の接続と区別するために使用する名前を入力します。たとえば、**PostgreSQL DB 1** を入力します。
9. **Description** は任意のフィールドで、この接続に関する便利な情報を入力します。たとえば、**Sample PostgreSQL connection that uses my login credentials.** を入力します。
10. **Save** をクリックし、作成した接続が利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できる接続として **PostgreSQL DB 1** が表示されます。

25.2. データベース接続でスキーマを指定する方法

Fuse Online では、データベース接続の作成時に、ユーザーインターフェイスでデータベースのスキーマを指定するよう要求されます。スキーマの指定方法はデータベースごとに異なります。以下の表は、各データベースに必要なものを表しています。

データベース	接続設定の例	注記
Apache Derby	接続 URL (JDBC URL): jdbc:derby:memory:testdb;create=true ユーザー名: sa パスワード: 任意の値 スキーマ: sample	テストの目的でのみ使用してください。デフォルトのユーザー名は sa で、パスワードは任意の値ですが、値を入力する必要があります。この設定例では、接続で namespace のような sample スキーマを使用します。たとえば、 sample.mytable で mytable テーブルを参照できます。

データベース	コネクション設定の例	注記
MS SQLServer	接続 URL (JDBC URL): jdbc:sqlserver://localhost:1433/sampledb ユーザー名:kurt パスワード:mypw スキーマ:sampledb	エクステンションの仕組み を利用してドライバをアップロードします。コネクションを作成します。ドライバをアップロードしていない場合は、コネクションの検証に失敗します。コネクション URL の最後と、 Schema フィールドに同じスキーマを指定する必要があります。
MySQL	接続 URL (JDBC URL): sql-connector.url=jdbc:mysql://mysql-host:3307/test?allowPublicKeyRetrieval=true&useSSL=false&useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDatetimeCode=false&serverTimezone=UTC ユーザー名: kurt パスワード: mypw スキーマ:	MySQL はスキーマをサポートしません。MySQL データベースへのコネクションを作成する場合は Schema フィールドを空白のままにしておきます。MySQL は、接続 (JDBC) URL のすべて設定を想定します。
Oracle DB	接続 URL (JDBC URL): jdbc:oracle:thin:@10.192.212.255:1521:xe ユーザー名:kurt Password:mypw スキーマ:	Fuse Online の エクステンションの仕組み を使用して Oracle データベースドライバをアップロードします。コネクションを作成します。ドライバがアップロードされていない場合は、コネクションの検証に失敗します。 スキーマはデータベース名です。Oracle の用語ではスキーマと呼ばれるこのデータベースインスタンスにアクセスするパーミッションが必要です。 Schema フィールドを空白のままにしておきます。 Connection フィールドで、このデータベースインスタンスの Oracle システム識別子 (SID) を参照する必要があります (この例では xe)。
PostgreSQL	接続 URL (JDBC URL): jdbc:postgresql://syndesis-db:5432/sampledb ユーザー名:kurt パスワード:mypw スキーマ:sampledb	接続 (JDBC) URL でデータベースを指定する必要があります。スキーマを使用する場合は、 Schema フィールドにスキーマを指定する必要があります。クエリーは、 database.schema.table 形式のテーブルを参照できます。参照がテーブル名のみを指定する場合、コネクションの設定時に指定したスキーマを最初に検索します。テーブルが見つからない場合、コネクションは指定のテーブルのパブリックリソースを検索します。詳細は、 PostgreSQL のスキーマドキュメント を参照してください。

データベース	コネクション設定の例	注記
Teiid	接続 URL (JDBC URL): jdbc:teiid: <vdb-name>@mm://localhost:31000 ユーザー名: any value Password: any value Schema:	Data Virtualization 機能などを使用して、仮想データベースがすでに作成済みである必要があります。接続 URL の <vdb-name> を仮想データベースの名前に置き換えます。Teiid への接続に対するセキュリティはまだ実装されていません。 Username および Password フィールドに任意の値を指定します。 Schema フィールドを空白のままにしておきます。このドライバーは Fuse Online ですでに利用可能です。



重要

Data Virtualization はテクノロジープレビューの機能です。テクノロジープレビュー機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、Red Hat は本番環境での使用は推奨しません。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能により、近日発表予定の製品機能をリリースに先駆けてご提供でき、お客様は開発プロセス時に機能をテストして、フィードバックをお寄せいただくことができます。Red Hat のテクノロジープレビュー機能のサポート範囲に関する詳細は、<https://access.redhat.com/ja/support/offerings/techpreview> を参照してください。

25.3. データベースレコードを取得してインテグレーションの実行をトリガー

SQL ステートメントまたは SQL ストアドプロシージャの呼び出し結果を基にしてインテグレーションの実行をトリガーするには、データベースコネクションをインテグレーションの最初のコネクションとして選択します。

前提条件

データベースコネクションが作成されている必要があります。

手順

1. Fuse Online パネルの左側にある **Integrations** をクリックします。
2. **Create Integration** をクリックします。
3. **Choose a connection** ページで、インテグレーションを開始するために使用するデータベースコネクションをクリックします。
4. **Choose an action** ページで、以下の1つを選択します。
 - **Periodic SQL invocation** は、指定した SQL ステートメントを定期的呼び出して、データを取得します。
 - **Periodic stored procedure invocation** は、選択または指定したストアドプロシージャを定期的呼び出して、データを取得します。
5. **Periodic SQL invocation** を選択した場合、**Query** フィールドに SQL **SELECT** ステートメント

または1つ以上のレコードを取得する他の標準のSQLステートメントを入力します。たとえば **SELECT * from my_db_table** を指定します。必要なデータが含まれるデータベーステーブルが存在する必要があります。

Periodic stored procedure invocation を選択した場合、**Procedure name** フィールドに必要なデータを取得するために呼び出すストアドプロシージャを選択または入力します。指定するストアドプロシージャが存在する必要があります。データベース管理者は、インテグレーションで使用する必要があるストアドプロシージャを作成しておく必要があります。

6. **Period** フィールドに整数を入力し、単位がミリ秒、秒、分、時間、または日であることを示します。たとえば、**5 minutes** を指定した場合、コネクションは指定のステートメントまたはストアドプロシージャを5分間隔で呼び出します。
7. **Next** をクリックします。

SQL エクステンションを使用する時の制限事項

Query フィールドに入力するSQLステートメントは、標準のSQLコンストラクトでなければなりません。Fuse Online は、Procedural Language/PostgreSQL (PL/pgSQL) または Oracle Procedural Language Extensions to SQL (PL/SQL) などのSQLエクステンションを認識または解析しません。ただし、ストアドプロシージャではSQLエクステンションを使用できます。

結果

Fuse Online はコネクションの検証を試みます。これには、指定のSQLステートメントの構文チェックや、ステートメントまたはストアドプロシージャターゲットデータの存在確認が含まれます。検証に成功した場合、Fuse Online は最初のコネクションをインテグレーションに追加します。検証に失敗した場合、問題に関するメッセージが Fuse Online に表示されます。必要に応じて入力を更新し、再試行します。

25.4. データを更新する SQL ステートメントのパラメータープレースホルダーおよび値

フローの途中にあるデータベースコネクションや、シンプルなインテグレーションを終了するデータベースコネクションは、データを更新するSQLステートメントを実行できます。データベースコネクションをインテグレーションに追加するとき、実行するSQLステートメントのプレースホルダーパラメーターを指定できます。実行中、入力パラメーターの値がコレクションにあるとコネクションはバッチ更新を実行できます。

SQL ステートメントでのパラメーターの指定

インテグレーションを作成または編集し、データベースを更新するコネクションを追加する場合、ランタイムにコネクションが実行するSQLステートメントにプレースホルダーパラメーターを指定したり、ランタイムに実行されるストアドプロシージャにプレースホルダーを含めることができます。以下に例を示します。

```
INSERT INTO TODO(task, completed) VALUES(:#param_1, :#param_2)
DELETE FROM TODO WHERE task LIKE :#param_3
```

これらのプレースホルダーの値を指定するには、フローのデータベースコネクションの前にデータマッピングステップを追加します。データマッピングステップで、適切なソースデータフィールドをターゲットデータフィールドにマップします。たとえば、ソースデータを **:#param_1**、**:#param_2**、および **:#param_3** ターゲットフィールドにマップします。[データマッピングステップの追加](#) を参照してください。

パラメーター値のコレクションでのバッチ更新

ランタイムにデータベースコネクションがデータを挿入、削除、または更新する SQL ステートメントを実行すると、実行された SQL ステートメントは多くの場合で1つ以上のプレースホルダーパラメーター (例: `:#task`) を指定します。この場合、パラメーターの単一セットまたはパラメーターセットのコレクションをコネクションの入力とすることができ、セットの各要素が対応するプレースホルダーパラメーターの値を定義します。

たとえば HTTP リクエスト (Webhook) や API プロバイダーインテグレーションの REST API サービスに送信されたリクエストなど、データベースコネクションへの入力はフローの前のコネクションから取得されます。入力がコレクションである場合、コネクションはバッチモードを使用してテーブルを更新します。たとえば、以下の SQL ステートメントについて考えてみましょう。

```
insert into todo (task) values (:#task)
```

入力コレクションに3つの値が含まれる場合、コネクションは値ごとに1つずつ、3つのレコード挿入します。ここで、2つのプレースホルダーパラメーターを持つ SQL ステートメントについて考えてみましょう。

```
insert into todo (task, completed) values (:#task, :#completed)
```

この SQL ステートメントを実行するコネクションへの入力で、データシェイプ仕様は単一の要素を指定できます。例を以下に示します。

```
{"task": "write some docs", "completed": 0}
```

代わりにコレクションを指定することもできます。例を以下に示します。

```
[{"task": "write doc", "completed": 1},  
 {"task": "publish doc", "completed": 0}]
```

コレクションを入力とする場合、コネクションはパラメーター値に各セットに対して1度挿入操作を実行します。この例では、コネクションはパラメーター値の各セットに1つずつ、2つのレコードをテーブルに追加します。

25.5. インテグレーションの途中またはインテグレーションを完了するためにデータベースにアクセス

インテグレーションでは、フローの途中でデータベースにアクセスしたり、シンプルなインテグレーションを終了するためにデータベースにアクセスすることができます。これには、データベースコネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- データベースコネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、コネクションの追加先のプラス記号をクリックします。Fuse Online が最後のコネクションを選択するよう要求する場合は、このステップをスキップします。

2. アクセスするデータベースのデータベースコネクションをクリックします。
 3. **Choose an action** ページで、以下の1つを選択します。
 - **Invoke SQL** は、指定の SQL ステートメントを実行してデータを操作します。
 - **Invoke stored procedure** は指定または選択したストアドプロシージャを呼び出してデータを操作します。
 4. **Invoke SQL** を選択した場合、**SQL Statement** フィールドで以下を行います。
 - 途中のコネクションでは、1つ以上のレコードを取得する SQL **SELECT** ステートメントを入力するか、1つ以上のレコードを操作する SQL **INSERT**、**UPDATE**、または **DELETE** ステートメントを入力します。データが含まれるデータベースが存在する必要があります。
 - 最後のコネクションでは、SQL **INSERT**、**UPDATE**、または **DELETE** ステートメントを入力し、1つ以上のレコードを操作します。
 - **Batch update** フィールドで、デフォルトの **No** を使用するか、**Yes** を選択します。**Batch update** の設定は、このアクションへの入力コレクションで、SQL ステートメントがレコードを挿入、削除、または更新する場合にコネクションの動作に影響します。デフォルトの動作 (**Batch update** が **No** の場合) では、コネクションは個別のオブジェクトのみを受け入れ、各オブジェクトに対して1度 SQL ステートメントを実行します。**Batch update** が **Yes** の場合、コネクションはコレクションをアクションへの入力として受け入れます。コネクションは SQL ステートメントを1度実行し、すべてのコレクションアイテムをバッチ更新操作への入力として使用します。
- Invoke stored procedure** を選択した場合、**Procedure name** フィールドに必要なデータを操作するために呼び出すストアドプロシージャの名前を選択または入力します。指定するストアドプロシージャが存在する必要があります。データベース管理者は、インテグレーションで使用する必要があるストアドプロシージャを作成しておく必要があります。
5. **Next** をクリックします。

結果

Fuse Online はコネクションの検証を試みます。これには、指定の SQL クエリーの構文チェックや、クエリーまたはストアドプロシージャターゲットデータの存在確認が含まれます。検証に成功した場合、Fuse Online でコネクションがフローに追加されます。検証に失敗した場合、問題に関するメッセージが Fuse Online に表示されます。必要に応じて入力を更新し、再試行します。

その他のリソース

[データを更新する SQL ステートメントのパラメータープレースホルダーおよび値](#)

25.6. プロプライエタリーデータベースへの接続

プロプライエタリー SQL データベースに接続する場合、以下の主なタスクを行う必要があります。

1. 開発者は、インテグレーションで接続するデータベースの JDBC ドライバーが含まれるライブラリーエクステンションを作成します。[JDBC ドライバーライブラリーエクステンションの作成](#) を参照してください。
2. 開発者はライブラリーエクステンション機能が含まれる **.jar** ファイルを提供します。
3. その **.jar** ファイルを Fuse Online にアップロードします。[カスタム機能の使用](#) を参照してください。

4. Fuse Online の **Database** コネクタを選択し、データベースの接続 URL を指定して、データベースへの接続を作成します。[データベース接続の作成](#) を参照してください。
5. インテグレーションで、接続をデータベースに追加します。[データベースレコードを取得してインテグレーションの実行をトリガー](#) または [インテグレーションの途中またはインテグレーションを完了するためにデータベースにアクセス](#) を参照してください。

第26章 TELEGRAM への接続

ビジネスユーザーは Telegram に接続するインテグレーションを作成できます。Telegram への接続は以下の1つを行うことができます。

- 指定の Telegram ボットがメッセージを受信するときにシンプルなインテグレーションの実行をトリガーします。インテグレーションはメッセージをフローの次のステップに渡します。たとえば、シンプルなインテグレーションがメッセージを受信した後に、製品名などのキーワードインスタンスを監視するフィルタステップがあることがあります。これらの製品名が含まれるメッセージを見つけた後、インテグレーションは Gmail コネクションまたは異なる Telegram コネクションの適切な連絡先に通知することができます。
- メッセージを特定の Telegram チャットに配信します。たとえば、この動作は、インテグレーションが FTP サーバーからファイルをダウンロードし、何らかの方法で処理する場合に便利です。プロセスに成功したことを Telegram チャットに通知する Telegram コネクションを追加できます。

インテグレーションで Telegram に接続するには、Telegram コネクションを作成した後に接続をインテグレーションフローに追加します。詳細は以下を参照してください。

- [「Telegram コネクションの作成」](#)
- [「メッセージの受信時に Telegram コネクションを追加してインテグレーションの実行をトリガー」](#)
- [「メッセージを Telegram チャットに送信するため Telegram コネクションを追加」](#)

26.1. TELEGRAM コネクションの作成

インテグレーションでは、Telegram コネクションは特定のチャットに送信されたメッセージを受信できます。作成したチャットボットはメッセージを Fuse Online に転送します。Telegram コネクションもメッセージを接続がアクセスできるチャットに送信することができます。以下で、同じ Telegram コネクションを使用できます。

- コネクションを使用してメッセージをチャットに送信する任意の数のインテグレーション。
- さらに、Telegram チャットからメッセージを受信するためにコネクションを使用する1つのインテグレーションで使用することができます。

そのため、コネクションが両方のインテグレーションで Telegram チャットメッセージを受信する場合、同じ Telegram コネクションを複数のインテグレーションで使用できません。これが Telegram の制限になります。

前提条件

- Telegram で、メッセージの取得元となるチャットのボットが作成されている必要があります。ヘルプが必要な場合は [Bots, an introduction for developers](#) を参照してください。
- Telegram BotFather からボット承認トークンを取得済みである必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能な接続を表示します。
2. 右上にある **Create Connection** をクリックし、Fuse Online コネクターを表示します。

3. **Telegram** コネクタをクリックします。
4. **Authorization Token** フィールドに、メッセージの受信元および送信先となるボットの Telegram 承認トークンを入力します。
5. **Next** をクリックします。
6. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Telegram for Product Mentions** を入力します。
7. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。
8. **Save** をクリックし、作成したばかりのコネクションを含む、利用可能なコネクションの更新済みリストを表示します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Telegram for Product Mentions** が表示されます。このコネクションを任意の数のコネクションに追加できます。

26.2. メッセージの受信時に TELEGRAM コネクションを追加してインテグレーションの実行をトリガー

インテグレーションを開始する Telegram コネクションは、コネクションが指定の Telegram チャットボットからメッセージを受信するたびにインテグレーションの実行をトリガーします。

前提条件

- Telegram コネクションが作成されている必要があります。
- シンプルなインテグレーションを作成することになります。Fuse Online はインテグレーションの開始方法の選択を要求します。

手順

1. インテグレーションの開始に使用する Telegram コネクションをクリックします。
2. **Read Messages** アクションを選択し、チャットボットに送信されたすべてのメッセージを受信します。
コネクション設定は、メッセージの受信元となるチャットボットを特定します。他の設定は必要ありません。

26.3. メッセージを TELEGRAM チャットに送信するため TELEGRAM コネクションを追加

インテグレーションでは、フローの途中でメッセージを Telegram チャットに送信したり、シンプルなインテグレーションを終了するために Telegram チャットを送信することができます。これには、Telegram コネクションをフローの途中に追加するか、シンプルなインテグレーションの最後のコネクションとして追加します。

前提条件

- Telegram コネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。または、Fuse Online で最後のコネクションを選択するよう要求されます。

手順

1. **Add to Integration** ページで、接続の追加先のプラス記号をクリックします。Fuse Online が最後の接続を選択するよう要求する場合は、このステップをスキップします。
2. フローに追加する Telegram 接続をクリックします。
3. **Send a Text Message** アクションを選択します。
4. **Chat Id** フィールドに、メッセージの送信先となる Telegram チャットの ID を入力します。このフィールドを空白のままにし、メッセージを受信する以前の Telegram 接続がある場合、Fuse Online は受信メッセージにあるチャット ID を使用します。

チャットの ID を取得するには以下を行います。

- a. Telegram で、ボットをメッセージの送信先となるチャットに追加します。BotFather 応答にはボットアクセストークンが含まれます。
- b. アクセストークンをクリップボードにコピーします。
- c. Telegram で、メッセージを直接そのチャットに送信します。メッセージの送信にボットは使用しないでください。
- d. 以下の形式で **curl** コマンドを呼び出し、チャットに送信されたメッセージを読み取ります。**bot-access-token** はクリップボードのアクセストークンに置き換えてください。
curl https://api.telegram.org/bot-access-token/getUpdates?offset=0

応答は JSON 形式になります。JSON **chat** オブジェクトにはチャット ID が含まれます。

5. **Next** をクリックして、接続をフローに追加します。

第27章 TWITTER への接続

インテグレーションで Twitter に接続するための主なステップは次のとおりです。

- 「Fuse Online を Twitter クライアントアプリケーションとして登録」
- 「Twitter コネクションの作成」
- 「Twitter コネクションを追加してインテグレーションの実行をトリガー」
- 「ダイレクトメッセージを送信する Twitter コネクションの追加」

Fuse Online 環境から Twitter へのアクセスを承認するため、Twitter 開発者アカウントが必要です。Twitter 開発者アカウントを持っていない場合は、<https://apps.twitter.com> にアクセスし、Twitter にログインしてから **Apply for a developer account** をクリックします。**you@gmail.com** のような個人アドレスではなく、**me@redhat.com** のような企業のアドレスを提供すると、迅速に開発者アカウントを取得できます。

27.1. FUSE ONLINE を TWITTER クライアントアプリケーションとして登録

インテグレーションで Twitter に接続するには、最初に Fuse Online 環境を Twitter にアクセスできるクライアントアプリケーションとして登録する必要があります。これにより、Twitter に接続するインテグレーションをいくつでも作成することができます。そのため、特定の Fuse Online 環境を 1 度だけ Twitter に登録する必要があります。

Fuse Online を Twitter クライアントアプリケーションとして登録できるのは、Fuse Online 環境ごとに 1 回のみです。しかし、各 Twitter コネクションは同じ登録を使用しますが、異なるユーザークレデンシャルを使用することができます。

前提条件

Fuse Online にログインしている必要があります。Fuse Online にアクセスできる URL は、Red Hat Fuse Online Evaluation プログラムのウェルカムメールメッセージにあります。

手順

1. Fuse Online で以下を行います。
 - a. 左側のパネルで **Settings** をクリックします。
 - b. **Settings** ページで、上部付近にあるコールバック URL の右側で  をクリックし、Fuse Online インストールのコールバック URL をクリップボードにコピーします。この手順の後半でこの URL が必要になります。
 - c. **Twitter** エントリーをクリックし、**Consumer API Key** および **Consumer API Secret Key** フィールドを表示します。
2. 別のブラウザータブで、[Twitter 開発者アカウント](#) にアクセスし、必要な場合はログインしてから以下を行います。
 - a. 右上で、開発者プロジェクトの名前の横にある下向きのチャレットをクリックし、**Apps** をクリックします。
 - b. 右上の **Create an app** をクリックします。

- c. **App name** フィールドに **My Fuse Online Client** などの名前を入力します。
 - d. **Application description** フィールドに、**For leveraging tweets** などのヒントを入力します。
 - e. **Website URL** フィールドに、この手順の最初でコピーした URL を貼り付け、URL の最後にある **api/v1/credentials/callback** を削除します。
 - f. **Enable Sign in with Twitter** を選択します。
 - g. **Callback URLs** フィールドに URL を再度貼り付けます。 **https://app-proj912876.7b63.fuse-ignite.openshiftapps.com/api/v1/credentials/callback** のようになるはずですが、
続くフィールドをいくつかスキップします。
 - h. **Tell us how this app will be used**には説明を 100 文字以内で追加する必要があります。 **This is a demo application so that I can evaluate how to integrate Twitter with my enterprise applications.** をコピーおよび貼り付けて対応することができます。
 - i. **Create** をクリックします。
 - j. 再度 **Create** をクリックし、開発者規約を読んだことを確認します。
 - k. 作成したアプリのページ上部にある **Keys and tokens** タブをクリックします。
 - l. コンシューマー API キーをコピーします。
3. Fuse Online の **Settings** ページに戻り、Twitter のコンシューマー API キーを Fuse Online の **Twitter Consumer API Key** フィールドに貼り付けます。
 4. Twitter の **Keys and tokens** タブに戻り、コンシューマー API シークレットキーをコピーします。
 5. Fuse Online の **Settings** ページに戻り、Twitter のコンシューマー API シークレットキーを Fuse Online の **Twitter Consumer API Secret Key** フィールドに貼り付けます。
 6. **Save** をクリックします。
 7. **Twitter** エントリーをクリックして折りたたみます。

27.2. TWITTER コネクションの作成

Twitter からデータを取得するインテグレーションを作成するには、最初に Twitter コネクションを作成する必要があります。Twitter コネクションを作成した後、任意の数のインテグレーションで使用できます。

前提条件

- Fuse Online にログインしている必要があります。Fuse Online にアクセスできる URL は、Red Hat Fuse Online Evaluation プログラムのウェルカムメールメッセージにあります。
- Fuse Online がウェブブラウザで開かれている必要があります。
- Fuse Online 環境が Twitter にアクセスできるアプリケーションとして登録されている必要があります。

- Fuse Online の **Settings** ページで登録した後に受け取った Twitter コンシューマー API キーおよびコンシューマー API シークレットキーが追加済みである必要があります。

手順

1. Fuse Online の左パネルで **Connections** をクリックし、利用可能なコネクションを表示します。
2. 右上の **Create Connection** をクリックし、利用可能なコネクタを表示します。コネクタは、1つ以上のコネクションを作成するために使用するテンプレートです。
3. **Twitter** コネクタをクリックします。
4. **Connect Twitter** をクリックし、Twitter の承認ページを表示します。承認ページを表示するには Twitter にログインする必要があります。
Connect Twitter が表示されない場合、Fuse Online 環境は Twitter クライアントアプリケーションとして登録されていません。[Fuse Online を Twitter クライアントアプリケーションとして登録](#) を参照してください。Twitter コネクションの作成時に Fuse Online 環境が Twitter クライアントアプリケーションとして登録されていない場合、Fuse Online は承認情報を要求する複数のフィールドを表示します。これらのフィールドに値を入力して Twitter コネクションを作成することはできますが、推奨されません。
5. **Authorize app** をクリックし、Fuse Online に戻ります。
6. **Name** フィールドに、このコネクションを別のコネクションと区別するために使用する名前を入力します。たとえば、**Twitter Connect 1** を入力します。
7. **Description** は任意のフィールドで、このコネクションに関する便利な情報を入力します。たとえば、**Sample Twitter connection that uses my Twitter login credentials.** を入力します。
8. **Save** をクリックし、作成したコネクションが利用可能になったことを確認します。上記の例の名前を入力した場合は、インテグレーションに追加できるコネクションとして **Twitter Connect 1** が表示されます。

27.3. TWITTER コネクションを追加してインテグレーションの実行をトリガー

シンプルなインテグレーションでは、特定のツイートやダイレクトメッセージを見つけたときにインテグレーションの実行をトリガーする Twitter コネクションを最初のコネクションとすることができます。

前提条件

- Twitter コネクションが作成されている必要があります。
- シンプルなインテグレーションを作成または編集することになります。Fuse Online は最初のコネクションを選択するよう要求します。

手順

1. インテグレーションを開始する Twitter コネクションをクリックします。インテグレーションが Twitter に接続するために選択したコネクションを使用する場合、Fuse Online はそのコネクションに定義されたクレデンシャルを使用します。

2. コネクションが実行するアクションを選択します。インテグレーションに追加した Twitter コネクションは選択したアクションのみを実行します。
- **Mention** は、コネクションのアクセスが許可されている Twitter アカウントの Twitter ハンドルが含まれるツイートを Twitter で監視します。コネクションはそのようなツイートを見つけると Fuse Online に返し、インテグレーションの実行がトリガーされます。
 - **Retrieve** は、Twitter コネクションのアクセスが許可されている Twitter アカウントに送信されるダイレクトメッセージを Twitter で定期的にポーリングします。コネクションはそのようなメッセージを見つけると Fuse Online に返し、インテグレーションの実行がトリガーされます。
 - **Search** は、指定した条件と一致するツイートを Twitter で定期的にポーリングします。コネクションはそのようなツイートを見つけると Fuse Online に返し、インテグレーションの実行がトリガーされます。

3. 選択したアクションを設定します。

- **Mention** を設定する必要はありません。
- **Retrieve**
 - **Count** のデフォルトは 100 で、内部の動作に影響するこの値を変更する必要はありません。パフォーマンスが低下する可能性があるため、50 未満の値を設定しないでください。
Retrieve アクションは、過去 30 日以内に受信された返送されていないすべてのダイレクトメッセージを常に返します。**Ignore direct messages previously found** を選択しないと、**Retrieve** アクションは過去 30 日以内に受信されたすべてのメッセージを返します。
 - **Delay** はポーリングの間隔です。Twitter はレート制限を適用するため、ポーリングを過剰に行わないようにします。デフォルトの間隔は 5 秒です。
 - **Ignore direct messages previously found** を選択すると、新しいメッセージのみが返されます。
- **Search**
 - **Delay** はポーリングの間隔です。Twitter はレート制限を適用するため、ポーリングを過剰に行わないようにします。デフォルトの間隔は 5 秒です。
 - **Ignore tweets previously found** を選択すると、新しいツイートのみが返されます。
 - **Query** は、コネクションが返すツイートを指定する Twitter 形式の検索式です。以下の表は、入力可能な内容を示しています。詳細は [検索演算子に関する Twitter ドキュメント](#) を参照してください。

クエリー	結果
My Product	My および Product が両方含まれるツイートを取得しますが、 My Product が必ず含まれるとは限りません。
"My Product"	My Product のインスタンスが含まれるツイートを取得します。

My OR Product	My または Product のインスタンス、またはそれぞれのインスタンスが含まれるツイートを取得します。
My -Product	My のインスタンスが含まれ、 Product のインスタンスが含まれないツイートを取得します。
#MyProduct	MyProduct ハッシュタグが含まれるツイートを取得します。

4. **Next** をクリックして接続をインテグレーションに追加します。

結果

インテグレーションに最初のコネクションが追加され、Fuse Online は最後のコネクションを選択するよう要求します。

27.4. ダイレクトメッセージを送信する TWITTER コネクションの追加

Twitter ユーザーにダイレクトメッセージを送信する Twitter コネクションをフローの途中に追加したり、シンプルなインテグレーションを終了するために追加することができます。ダイレクトメッセージは、指定ユーザーのみに送信されるプライベートメッセージのことです。

前提条件

- Twitter コネクションが作成されている必要があります。
- フローを作成または編集することになり、Fuse Online でインテグレーションに追加するよう要求されます。Fuse Online は、インテグレーションを追加するよう要求するか、シンプルなインテグレーションでは最後のコネクションを選択するよう要求します。

手順

1. **Add to Integration** ページで、Twitter コネクションを追加する場所のプラス記号をクリックします。
2. インテグレーションに追加する Twitter コネクションをクリックします。インテグレーションが Twitter に接続するために選択したコネクションを使用する場合、Fuse Online はそのコネクションに定義されたクレデンシャルを使用します。
3. **Send** アクションを選択します。
4. **Default Message** フィールドには、データマッピングステップが前のステップからこの Twitter コネクションにダイレクトメッセージをマップしない場合にコネクションが送信するメッセージを入力します。この Twitter コネクションの設定時にメッセージを入力する必要がありますが、前のステップからマップするメッセージはここで指定するメッセージよりも優先されます。
5. **User** フィールドに、ダイレクトメッセージの宛先となるユーザーの Twitter ハンドルを入力します。たとえば、**Aslan** と **@Aslan** はいずれも適切です。
6. **Next** をクリックして接続をインテグレーションに追加します。

結果

インテグレーションビジュアライゼーションの途中または最後にコネクションが表示されます。